United States District Court
For the Northern District of California

IN THE UNITED STATES DISTRICT COURT

FOR THE NORTHERN DISTRICT OF CALIFORNIA

NETWORK APPLIANCE INC,

Plaintiff,

v.

SUN MICROSYSTEMS INC,

Defendant.

______________________________________/

No. C-07-06053 EDL

ORDER CONSTRUING CLAIMS

On August 27, 2008, the Court held a claim construction hearing to construe the disputed terms of United States Patent Numbers 5,925,106 (the "'106 patent"), 5,459,857 (the "'857 patent"), 5,749,095 (the "'095 patent"), 6,873,630 (the "'630 patent") (collectively, "Sun patents"), and U.S. Patent Nos. 5,819, 292 (the "'292 patent"), 6,892, 211 (the "'211 patent"), and 7,200,715 (the "'715 patent") (collectively, "NetApp patents") pursuant to Markman v. Westview Instruments, Inc., 517 U.S. 370 (1996). Having read the parties' papers and carefully considered their arguments and the relevant legal authority, the court hereby rules as follows.

I. BACKGROUND

On September 5, 2007, Network Appliance, Inc. ("NetApp") filed its Complaint in the Eastern District of Texas, alleging that Sun Microsystems, Inc. ("Sun") infringed and is infringing, directly and indirectly under 35 U.S.C. § 271, certain patents, including the '292 patent, the '211 patent, and the '715 patent, by making, using, selling, or offering for sale data processing systems and related software, primarily Zettabyte File System ("ZFS"). NetApp further seeks a declaratory judgment that certain patents owned by Sun are each not infringed, are invalid and/or are unenforceable, and it seeks a permanent injunction to prevent future infringements as well as damages adequate to compensate it for Sun's past infringement and trebling of damages pursuant to

United States District Court
For the Northern District of California

35 U.S.C. § 284.

On October 25, 2007, Sun filed its Answer and Counterclaim, denying the material allegations of the Complaint and asserting a number of affirmative defenses and counterclaims. Sun denies infringing any of the NetApp Patents and alleges that NetApp infringes a number of its patents, including its four patents at issue in this claim construction. Sun further seeks a declaratory judgment that the NetApp Patents are each not infringed, are invalid and/or are unenforceable, as well as a permanent injunction to prevent future infringements as well as damages adequate to compensate it for NetApp's past infringement and trebling of damages pursuant to 35 U.S.C. § 284.

The case was transferred to this District on November 30, 2007. On December 7, 2007, NetApp filed its Reply to Sun's Answer and Counterclaims, denying the material allegations of Sun's Answer and Counterclaim and asserting a number of affirmative defenses and counterclaims.

The parties now seek construction of fourteen disputed terms and/or phrases, which are contained in various claims in the seven patents.

II. DISCUSSION

A. Legal Standard

In construing claims, the court must begin with an examination of the claim language itself. The terms used in the claims are generally given their "ordinary and customary meaning." See Phillips v. AWH Corp., 415 F.3d 1303, 1312-13 (Fed. Cir. 2005); see also Renishaw PLC v. Marposs Societa' per Azioni, 158 F.3d 1243, 1248 (Fed. Cir. 1998) ("The claims define the scope of the right to exclude; the claim construction inquiry, therefore, begins and ends in all cases with the actual words of the claim."). This ordinary and customary meaning "is the meaning that the terms would have to a person of ordinary skill in the art in question at the time of the invention...". Phillips, 415 F.3d at 131. A patentee is presumed to have intended the ordinary meaning of a claim term in the absence of an express intent to the contrary. York Products, Inc. v. Central Tractor Farm & Family Ctr., 99 F.3d 1568, 1572 (Fed. Cir. 1996).

Generally speaking, the words in a claim are to be interpreted "in light of the intrinsic evidence of record, including the written description, the drawings, and the prosecution history, if in evidence." Teleflex, Inc. v. Ficosa North Am. Corp., 299 F.3d 1313, 1324-25 (Fed. Cir. 2002)

United States District Court
For the Northern District of California

(citations omitted); see also Medrad, Inc. v. MRI Devices Corp., 401 F.3d 1313, 1319 (Fed. Cir. 2005) (court looks at "the ordinary meaning in the context of the written description and the prosecution history"). "Such intrinsic evidence is the most significant source of the legally operative meaning of disputed claim language." Vitronics Corp. v. Conceptronic, Inc., 90 F.3d 1576, 1582 (Fed. Cir. 1996).

With regard to the intrinsic evidence, the court's examination begins, first, with the claim language. See id. Specifically, "the context in which a term is used in the asserted claim can be highly instructive." Phillips, 415 F.3d at 1314. As part of that context, the court may also consider the other patent claims, both asserted and unasserted. Id. For example, as claim terms are normally used consistently throughout a patent, the usage of a term in one claim may illuminate the meaning of the same term in other claims. Id. The court may also consider differences between claims as a guide in understanding the meaning of particular claim terms. Id.

Second, the claims "must [also] be read in view of the specification, of which they are a part." Id. at 1315. When the specification reveals a special definition given to a claim term by the patentee that differs from the meaning it would otherwise possess, the inventor's lexicography governs. Id. at 1316. Indeed, the specification is to be viewed as the "best source" for understanding a technical term, informed as needed by the prosecution history. Id. at 1315. As the Federal Circuit stated in Phillips, the specification is "the single best guide to the meaning of a disputed term," and "acts as a dictionary when it expressly defines terms used in the claims or when it defines terms by implication." 415 F. 3d at 1321.

Limitations from the specification, however, such as from the preferred embodiment, cannot be read into the claims absent a clear intention by the patentee to do. Teleflex, 299 F.3d at 1326 ("The claims must be read in view of the specification, but limitations from the specification are not to be read into the claims.") (citations omitted); CCS Fitness, 288 F.3d at 1366 ("a patentee need not describe in the specification every conceivable and possible future embodiment of his invention."); Altiris v. Symantec Corp., 318 F.3d 1363, 1372 (Fed. Cir. 2003) ("resort to the rest of the specification to define a claim term is only appropriate in limited circumstances").

"[T]here is sometimes a fine line between reading a claim in light of the specification, and

reading a limitation into the claim from the specification. . . . [A]ttempting to resolve that problem in the context of the particular patent is likely to capture the scope of the actual invention more accurately than either strictly limiting the scope of the claims to the embodiments disclosed in the specification or divorcing the claim language from the specification." Decisioning.com, Inc. v. Federated Dep't Stores, Inc., 527 F.3d 1300, 1308 (Fed. Cir. 2008) (citations omitted). There is therefore "no magic formula or catechism for conducting claim construction," and the court must "read the specification in light of its purposes in order to determine whether the patentee is setting out specific examples of the invention to accomplish those goals, or whether the patentee instead intends for the claims and the embodiments in the specification to be strictly coextensive." Id. (internal citations omitted).

Finally, as part of the intrinsic evidence analysis, the court "should also consider the patent's prosecution history, if it is in evidence." Phillips, 415 F.3d at 1317. The court should take into account, however, that the prosecution history "often lacks the clarity of the specification" and thus is of limited use for claim construction purposes. Id.

In most cases, claims can be resolved based on intrinsic evidence. See Vitronics, 90 F.3d at 1583. Only if an analysis of the intrinsic evidence fails to resolve any ambiguity in the claim language may the court then rely on extrinsic evidence, such as expert and inventor testimony, dictionaries, and learned treatises. See Vitronics, 90 F.3d at 1583 ("In those cases where the public record unambiguously describes the scope of the patented invention, reliance on any extrinsic evidence is improper"). "Within the class of extrinsic evidence, the court has observed that dictionaries and treatises can be useful in claim construction." Phillips, 415 F.3d at 1318. While expert testimony can be useful to a court for a variety of purposes, conclusory assertions by experts are not useful to a court. Id. The court generally views extrinsic evidence as less reliable than the patent and its prosecution history in determining how to read claim terms, even if though consideration is within the court's sound discretion. See id. at 1318-19.

B. Construction of Disputed Terms and Phrases

The parties dispute fourteen terms and/or phrases in the seven patents. The first seven disputed terms and/or phrases are contained in the Sun patents, and the latter seven terms are

contained in the NetApp patents.

Patent '106

This patent concerns an invention for obtaining and displaying information about a server on a network. It describes a mechanism for helping a user of the internet better understand the source of a webpage or server by adding a descriptive reference or parenthetical phrase after the web address on the user's computer screen to help the user of the internet identify the owner of the website or server. For example, this invention would add the parenthetical "Northern District of California" after the Court's website www.cand.uscourts.gov. The patent describes the invention within the context of a preferred embodiment using a world wide web browser and server, but notes that the internet browser and servers are representative of the technology. See '106 Patent at 1:31-36.

1. "Domain Name"

NetApp's Proposed Construction	Sun's Proposed Construction
A third-party-approved name of a website on the Internet, i.e. a registered domain name	A name that has a numerical IP address associated with it

The key issue here is whether domain name is limited to a registered name of an internet website, or is more broadly defined. To begin its analysis, the court first turns to the claims themselves. The term "domain name" appears in each independent claim of the '106 patent, claims 1, 9, 14, 22, 27, and 35. As Sun notes, the independent claims containing the term "domain name" describe systems and methods for providing server information to a client over a network. See, e.g., claims 1 ("provide access to a network"), 9 ("A server apparatus having . . . access to a network). In contrast, some dependent claims add the use of the www, urls, and http. See, e.g., claim 5 ("access mechanism further comprises a world wide web (www) browser apparatus configured to use . . . http to utilizing a uniform resource locator (url)", claim 11 (requires usage of www, http, and urls). Accordingly, it appears that NetApp's proposed construction, which limits domain names to registered internet domain names and would therefore imposes www, url, and http requirements in the independent claims, may render much of the language in the dependent claims superfluous. Such a reading would violate the principle of claim differentiation. See Uniroyal Inc. v. Rudkin-

Wiley Corp., 837 F.2d 1044, 1054-55 (Fed. Cir. 1988). However, it is not entirely clear from a plain reading of the claims whether the dependent claims are rendered completely superfluous, as the internet is broader than the world wide web, and encompasses other public network applications. See Almeroth Decl. at 10. Therefore, because the claim language is not dispositive, the court turns to the specification for further guidance.

The specification explains that “domain names are requested by maintainers of the website and are approved by a third party.” ’106 Patent at 1:56-57. In addition, the figures in the patent are cutouts of Netscape, a program known to look up websites on the internet based on domain names. See ’106 Patent at 5:15-21; Almeroth Decl. at 6. While this language at first blush may seem to suggest that domain names are limited to their use in websites and require third party approval, as Sun notes, this portion of the specification is discussing the www preferred embodiment, which the patent explicitly states is nonexclusive. See ’106 Patent at 1:32-37 (“Although the invention covers information access and information provider apparatus, www browser and www server applications are representative of the technology. As such, the majority of this application describes the invention within the context of a preferred embodiment utilizing www browser and www server applications.”); 4:24-29 (“Although the invention covers information access and information provider apparatus, methods, and computer program products, www browser and www server apparatus and applications are representative of the technology.”). Therefore, importing such definitions into the claim terms would improperly import limitations from the preferred embodiment.

The summary of the invention provides further evidence that the term should not be limited to the www and internet. It describes the invention as applying broadly to networks, servers, and client computers, which are not limited to the internet and www. ’106 Patent at 2:23-59. In addition, the operation of the invention does not always require third party approval or registered domain names. For instance, the patent explains that server identification data can be retrieved using a conventional database that associates domain names with server identification data. ’106 Patent at 7:56-8:4. This reference in the specification is consistent with Sun’s extrinsic expert testimony, which, while less important to the analysis, provides further support for the

United States District Court
For the Northern District of California

understanding that such a database need not be maintained by a third party or use only registered domain names. Kaliski Decl. ¶ 15.

NetApp argues that the references incorporated by the patent support its construction of the "domain name" term, as they describe the domain name by defining the host component of the URL as a legal internet host domain name or as a fully qualified domain name of a network host. NetApp Op. Brief at 6 n.2; Almeroth Decl. at 7. However, these definitions are limited to the context of the preferred embodiment and consequently are not persuasive. They also do not specifically refer to a registration requirement and are limited to "domain name systems" functions. Walter Decl., Ex. K § 2.1. Consequently, such definitions are not particularly relevant here.

In analyzing the specification as a whole, therefore, while NetApp's construction is supported by some direct language in the specification, that language refers to domain names only in the context of websites, a limitation which cannot be read into the term "domain name" in all contexts of the patent. In addition, at the hearing, NetApp acknowledged that its proposed construction effectively excludes private networks. This exclusion is improper, as the invention refers broadly to networks, and unique identifying information might be desirable on large private networks, such as may be maintained by businesses. According to Sun's expert's unrebutted testimony, for example, it is common for private networks to use domain names that are not third party approved or registered. See Kaliski Decl. ¶ 13. Imposing a uniqueness requirement for domain names and for IP addresses, however, would not exclude private networks or other embodiments of the invention, and Sun has not argued otherwise. The specification notes that the IP addresses and domain names must be unique. Id. at 1:57-58. While this is in the context of the preferred embodiment, logically speaking, it would be impossible to provide descriptive information about the server identified by the domain name if the domain name were not unique in any way. The specification, therefore, indicates that a uniqueness requirement should be incorporated into the definition of domain name.

The parties also rely on the patent prosecution history in support of their constructions, which is less important and less reliable in claim construction than the specification. See Phillips, 415 F.3d at 1317. The applicant explicitly defined "domain name" as a term "used in accordance

with standard usage in the field. It is provided to a nameserver to return an IP address." Walter Decl., Ex. E at 6 (Amendment A, Dec. 22, 1997). A nameserver is a mechanism for associating names with IP addresses. Id. at 4. These statements do not by themselves support NetApp's construction. NetApp notes, however, that the patentee also stated with regard to other information, that "unlike domain names, this additional information need not be unique nor approved by a third party." Id. at 3, 7. However, this latter statement was not directed to the terms or claims at issue. The explicit definition set forth for "domain name" in the prosecution history is more relevant than the inventor's comments about domain names regarding different patent claims. Thus, the prosecution history is too ambiguous to overcome the teachings of the claim language and the specification.

As to the parties' other extrinsic evidence, NetApp contends that the parties' dictionary definitions confirm that domain names are registered with third parties, as identifying the owner of the domain name only makes sense if the domain name is registered with a third party. See, e.g., Webster's New World Dictionary of Computer Terms, 8th Ed. (2000) ("domain name: in the system of domain names used to identify individual internet computers, a single word or abbreviation that makes up part of a computer's unique name") and Microsoft Press Computer Dictionary, 3d Ed., (1997) (defining domain name to mean "an address of a network connection that identifies the owner of that address in a hierarchical format: server.organization.type")). While these definitions might imply that internet addresses have owners, as Sun points out, there is nothing in these definitions that requires third party approval or registration. At the same time, Sun's own proffered Webster's definition requires that the domain name makes up part of a computer's *unique* name, which supports the specification's focus on this requirement. See '106 Patent at 1:57-58.

The parties discuss additional extrinsic evidence, none of which is persuasive given the intrinsic evidence discussed above. For example, while Professor Almeroth opines for NetApp that the term domain name is understood in the art to refer to an address of a computer on the internet, where the address is approved by a third party and stored in the internet's registry of domain names, this is not persuasive given the language used in the specification to encompass networks more broadly, as discussed above. Almeroth Decl. at 5:8-10. Sun also argues that NetApp's regular use

United States District Court
For the Northern District of California

of domain names to describe a "name that has a numerical IP address associated with it" and is not registered or third party approved, for example, in documentation describing the accused products, supports Sun's proposed construction. See Williamson Decl., Ex. A at 101, 103, 104, 105, and 107 (referring to NIS domain names, which according to Sun's expert, need not be third party approved); Kaliski Decl. ¶ 16. However, this extrinsic evidence is fairly attenuated and, as NetApp notes, the use of the phrase "NIS domain name" may just as easily be said to confirm that "domain name" by itself has a different meaning from NIS domain name.

Finally, NetApp cites Federal Circuit cases from 2003-2005 that discuss domain names as being premised on uniqueness of ownership and a centralized system of recording title, which indicate that they are registered. See Resonate Inc. v. Alteon Websystems, Inc., 338 F.3d 1360, 1361-62 (Fed. Cir. 2003) ("Every web page is identified by a unique Uniform Resource Locator (URL). . . . Every web site has a home page, which is identified by a URL and is the first document users see when they first connect to the web site. Also associated with each web site is a domain name, usually part of the URL."); In re Oppedahl & Larson LLP, 373 F.3d 1171, 1176-77 (Fed. Cir. 2004) ("The simple fact that domain names can only be owned by one entity does not of itself make them distinctive or source identifying."); and Teva Pharmaceuticals USA, Inc. v. Pfizer, Inc., 405 F.3d 990, 997 (Fed. Cir. 2005). However, the application for the '106 patent was filed in April 1996, over seven years before these cases were decided, so these cases are not extrinsic evidence of what one of ordinary skill in the art understood at the relevant time. In addition, none of these cases provide strong evidence that the term is limited to registered internet addresses. And while 15 U.S.C. § 1127, a statute regarding trademarks, not patents, defines domain name as "any alphanumeric designation which is registered with or assigned by any domain name registrar, domain name registry, or other domain name registration authority as part of an electronic address on the Internet," this definition was not enacted until November 1999. Nor is there any showing that one of ordinary skill in the art would know of the legislation.

In sum, and for all the reasons set forth above, the court adopts Sun's proposed construction of "domain name," modifying that term to include a uniqueness requirement and construes the term as: "a unique name that has a unique numerical IP address associated with it."

United States District Court
For the Northern District of California

2. "Server Identification Data"

NetApp's Proposed Construction	Sun's Proposed Construction
Human-friendly information identifying a specific web server designed not to be intimidating to inexperienced users of the world wide web[1]	Information that uniquely identifies one server from other servers and can be seen by a user.

NetApp's construction limits server identification data to that describing web servers to users of the world wide web, and further limits the term to human friendly information. The term "server identification data" appears in claims 1, 4, 9-10, 14, 17-18, 23-24, 27, 30, and 35-36 of the '106 patent. In claim 1, the "said server identification data including descriptive information about a server" is contained in the invention, along with "a display mechanism configured to display said server identification data on said display device." This claim language thus renders Sun's proposed construction somewhat redundant. For example, Sun's definition notes that the data can be seen by a user, but the separate claim terms about displaying said data on a display device already describes that function on its own.

The parties largely rely on the specification and prosecution history to construe this term. The specification explains that displaying "human-friendly server identification information to a user to better indicate the origin of the information" is the goal of the invention. '106 Patent at 1:15-18. The specification explains that the information is needed to help inexperienced users avoid becoming disoriented and intimidated when surfing on the world wide web. Id. at 1:45-47, 1:60-2:3. The specification gives the example in which the server's identification data includes the text string "Sun Microsystems" as descriptive information, which is enclosed in parentheses and separated from the URL by a space. Id. at 5:40-44. As noted above, the patent also specifically disclaims the www example as being the sole and exclusive focus of the invention, instead couching it as the context of a preferred embodiment, so importing an explicit web limitation into the claim term does

[1] At the hearing, without prior notice to Sun or the Court, NetApp proposed the following revised construction: "Information readily understood by an inexperienced user of the Internet that describes the identity of a specific server."

not seem warranted.[2] Id. at 1:32-39.

The broad purpose of the invention is, however, to provide "human-friendly server identification information to a user to better indicate the origin of the information," and such language describing "features of the present invention" limits a patent's scope. See Verizon Services Corp. v. Vonage Holdings Corp., 503 F.3d 1295, 1308 (Fed. Cir. 2007) ("When a patent thus describes the features of the 'present invention' as a whole, this description limits the scope of the invention."). It is true that if a patent refers to the present invention and preferred embodiment interchangeably, the use of the "present invention" language does not limit a patent's scope. Karlin Tech., Inc. v. Surgical Dynamics, Inc., 177 F.3d 968, 973 (Fed. Cir. 1999). However, the patent does not do so here; instead, the inventor referred to "this invention" when broadly discussing the purpose of "the invention" in the field of the invention section. See '106 Patent at 1:15-18. In addition, claim terms should be construed in a manner consistent with the "clear purpose of the invention." Cordis Corp. v. Medtronic Ave., Inc., 511 F.3d 1157, 1179 (Fed. Cir. 2008) (citation omitted); see also CVI/Beta Ventures v. Tura LP, 112 F.3d 1146, 1160 (Fed. Cir. 1997) ("In construing claims, the problem the inventor was attempting to solve, as discerned from the specification and the prosecution history, is a relevant consideration.").

Sun counters that the summary of the invention section of the specification broadly describes the invention as a way of providing server specific information to a computer user. '106 Patent at 2:24-28. The server identification data is obtained using a domain name and can be displayed using a display device, and includes "descriptive information about the server identified by the domain name." Id. at 2:28-31. Sun argues that this language directly supports its proposed claim construction. However, because the claims themselves refer to "server identification data including descriptive information about a server and accessible by use of said domain name," for example in claim 14, Sun's definition renders the other claim terms superfluous and repetitive, indicating that "server identification data" has a more specific meaning. In sum, the specification supports NetApp's requirement of human-friendly information, but not its reference to inexperienced users of

---

[2] NetApp's revised construction omits reference to the www, but still refers to users of the internet.

United States District Court
For the Northern District of California

the world wide web, as such limitations describe the preferred embodiment only.

Turning to the patent prosecution history, the inventor stated that the invention "addresses the problem of identifying and/or describing a server site to a human user by providing additional information about the server to the user," distinguishing other forms of information like domain names, hypertext links, URLs, and bookmarks. Walter Decl., Ex. E at 3-4. The prosecution history explicitly defines "server identification information" as a term "used throughout the application to mean descriptive information about a server that can be seen by the user – in particular text [citations to text omitted]. This descriptive information is not an IP address that is returned by a nameserver." Id. at 5-6. From the prosecution history, it is clear that the term "server identification data" excludes IP addresses, domain names, hypertext links, URLs, and bookmarks. NetApp argues that Sun's proposed construction must be wrong, because it would broadly cover IP addresses and URLs, which were specifically disclaimed in the file history. At the hearing, Sun conceded that IP addresses, domain names, and the like, in and of themselves did not constitute server identification data. While Sun argued that the term "server identification data" could still include such items, this distinction is meaningless: if such items cannot constitute the data, then it must be whatever else is provided, along with the IP address, and the like, that constitutes the identification data. The clear purpose of the invention – to provide more understandable, less technical information identifying the server – is supported by the unambiguous exclusion from the definition of IP addresses returned by a nameserver, domain names, hypertext links, URLs, and bookmarks.

The intrinsic evidence, therefore, teaches that the "server identification data" excludes IP addresses, domain names, hypertext links, and the like, as it is descriptive data that is more user-friendly. In sum, and for all the reasons set forth above, the court adopts a combination of the parties' proposed constructions that omits limitations to the world wide web, but includes the user-friendly information limitation, which distinguishes the invention from the prior art, and construes "server identification data" as: "information that uniquely identifies one server from other servers, that is more human user-friendly than the server's IP address, domain name, URL, or hypertext link." This definition stays true to the intrinsic evidence and addresses Sun's concern that by itself using the phrase "human-friendly information" may not be sufficiently clear to assist the jury.

Patent ’857

This patent is directed to a fault tolerant data storage system designed to minimize the risk of failure. The patent does so through redundancy – that is, storing data on two separate storage subsystems, such that the failure of a single subsystem would not lead to the loss of data. The storage control units of both data storage subsystems are synchronized to maintain identical images of both subsystems, so that the data records are stored in available memory on both storage subsystems. Data is exchanged over the data link to maintain consistency in the two sets of information. See ’857 Patent Abstract.

3. “In response to writing a data record to said one redundancy group”/”Responsive to writing a data record to one of said redundancy groups”

4. “In response to the receipt of a stream of data records from said data processor”/“Responsive to the receipt of a stream of the data records from said data processor”

The parties address terms three and four together, and the Court will do the same. The following are the parties’ constructions for the two phrases, respectively:

NetApp’s Proposed Constructions	**Sun’s Proposed Constructions**
After and in reaction to the writing of a data record to a single redundancy group; After and in reaction to the receipt of data records from a processor	In response to writing the data record to memory associated with the one redundancy group/Responsive to writing the data record to memory associated with one of the redundancy groups; Sun contends that the latter phrase does not require construction.

To begin its analysis, the court turns to the claims themselves. These phrases appear in claims 6 and 11 of the ’857 patent. Claim 6 states “transmitting, in response to writing a data record to said one redundancy group, said written data record to the other of said data storage subsystems via said data link to maintain duplicate data records in both said data storage subsystems.” NetApp argues that the claim language on its face requires that the transmission of a written data record from one data storage subsystem to another occur only after the record is written to a single redundancy

group,[3] because "in response" to something means "after" according to its plain language. NetApp also argues that the claim language further requires that the written data record be the thing that is transmitted to the other redundancy group, and that use of the past-tense of "written" requires that the writing be completed before the transmission.

It is true that "in response to" may mean "after." More importantly, the inclusion of the phrase "written data record" weighs heavily in favor of NetApp's interpretation that the transmission occurs after the complete writing of the data record, as the past tense term "written" has a plain meaning, indicating that the data record has already been written to a redundancy group before it is transmitted.

Turning to the specification, Sun argues that NetApp's construction fails to include the preferred embodiment, in which the writing occurs after the step of transmitting data to the other storage subsystem. The '857 patent describes a single preferred embodiment for writing duplicate data records through data storage subsystems.[4] The undisputed sequence is as follows. First, the host processor transfers the data record to the first data storage subsystem. '857 Patent at 11:51-54. Then, the first data storage subsystem writes the data record into its cache memory. Id. at 11:54-62. Then, the first data storage subsystem transfers the data record to the second data storage subsystem. Id. at 11:62-12:1. Finally, the data record is written from cache memory to disk. Id. at 12:3-9. According to Sun, because the data record is transmitted from the first data storage subsystem to the second data storage subsystem after it has been written to cache memory, but before it is written to a redundancy group of disks, the data storage subsystem can notify the processor that a write

[3] As explained in more detail below, "written to a redundancy group" means written to disk and cannot mean written to cache memory.

[4] NetApp also argues somewhat cursorily that the patent may disclose additional possible embodiments, as the patent describes both the writing step and transmitting step at one point without specifying any particular sequence for performance of these steps. '875 Patent at 4:39-50. That portion of the patent states: "Host processors 101, 121 transmit data record write and read requests to storage units 105 and 107 in conventional manner. The one of storage control units 105 and 107 that receives these requests . . . communicates with its associated disk controller 112 and disk drives 109 to execute the write and read requests. In addition, in response to a received data record write command, storage control unit 105 transmits the received data record over data link 106 to storage control unit 107 to maintain identical virtual device images in both storage control units 105 and 107 and identical data records in data storage subsystems 1,2." But this part of the specification does not disclose a different sequence from the preferred embodiment discussed above. Rather, it is simply silent as to timing, and lacks the specificity to constitute an alternative embodiment.

United States District Court
For the Northern District of California

operation is complete when a data record is stored in cache memory, even if the data has not been transferred to disk. ’857 Patent at 11:62-12:7. This frees a host processor from waiting until data has been written to disk and allows it to perform other tasks to increase performance and improve efficiency. Id. at 6:63-7:1.

NetApp, however, argues that not all claims must cover the preferred embodiment, especially where, as here, the patent has 54 claims including 16 independent claims, and only one detailed description of the preferred embodiment. Generally, a claim term should not be interpreted “in a way that excludes the preferred embodiment from the scope of the invention. . . Specifically, [the Federal Circuit] has cautioned against interpreting a claim term in a way that excludes disclosed embodiments, when that term has multiple ordinary meanings consistent with the intrinsic record.” Helmsderfer v. Bobrick Washroom Equip., Inc., 527 F.3d 1379, 1383 (Fed. Cir. 2008). In Helmsderfer, the appellant argued that the district court’s construction of the term was erroneous because it excluded both the preferred embodiment *and every illustrated embodiment* from the two particular claims at issue. The Court, however, noted that every single claim need not cover the preferred embodiment in the patent where at least some claims covered that embodiment. See 527 F.3d at 1383. Moreover, as Helmsderfer recognized, the claim term should only be construed to cover the preferred embodiment if one of the ordinary meanings consistent with the intrinsic record so permits. Cf. Sinorgchem Co. v. ITC, 511 F.3d 1132, 1138 (Fed. Cir. 2007) (“Where, as here, multiple embodiments are disclosed, we have previously interpreted claims to exclude embodiments where those embodiments are inconsistent with unambiguous language in the patent’s specification or prosecution history.”). Here, “written” does not have an alternative ordinary meaning.

NetApp does have a high hurdle to overcome to support its construction, as claims 6 and 11, if construed as NetApp proposes, would not encompass any disclosed embodiment. However, in Helmsderfer, the Court noted that some claims need not cover *any* of the embodiments if other claims cover those embodiments. This reasoning seems particularly applicable here, where there are so many claims and only one disclosed embodiment. In addition, the specific sequence of the response is not required in every claim in this patent. For example, claim 31 covers a data storage system comprising in part a “means in said first data storage control operable in response to the

receipt of said data record over one of said channel interface paths for transmitting said data record over a data link transmission path to said second data storage control" and does not refer to transmission of a written data record. '857 Patent at 20:37-41. Other claims are similar. See, e.g., id. at Claims 24, 29, and 33.

Here, the claim language is clear and unambiguous. Construing the term according to Sun's construction to include the preferred embodiment would result in a construction that is contrary to any ordinary meaning of the claim language. In order to have its construction cover cache memory, Sun inserts the language "memory associated with" said one redundancy group into the claim term. However, this improperly rewrites the claim to salvage it and does so in a manner that is inconsistent with the specification. The claim language does not refer to transmitting a written data record in response to writing said record to cache memory. Further, the patent makes clear that a redundancy group consists of disk drives. See '857 Patent at 12:9 (referring to "redundancy group of the disk drives"); 2:5-11 ("data storage subsystem consists of a plurality of small form factor disk drives which are configured into a plurality of redundancy groups"); 3:2-9 ("plurality of disk drives . . . are configured into a plurality of variable size redundancy groups"). Inserting the term "memory associated with" before the reference to redundancy group creates a construction at odds with the plain and consistent definition in the specification of a redundancy group as "a redundancy group consists of N+M disk drives." Id. at 7:18. Sun's proposed construction would take the plain language of the claim, transmitting in response to writing a data record to a redundancy group (or disk drives), and rewrite it into transmitting in response to writing a data record to cache memory associated with those disk drives. But cache memory cannot be equated with or vaguely attached to disk drives. For example, the specification states that "[a] virtual track is staged from a redundancy group into cache," which shows that the two are separate things. Id. at 3:65-66. It also states that a storage control unit writes the data record from *its* cache memory into a selected redundancy group of the disk drives, showing that cache memory pertains to the storage control unit and is separate from the redundancy group. See id. at 12:7-9. For these reasons, Sun's explanation that the term "written data record" refers to data that has been written to cache memory, but not to disk, is directly contrary to the plain language of the claim and to the specification.

The same reasoning applies to construction of the term "in response to the receipt of a stream of data records," which contains parallel language and indicates that the selection happens after the receipt of the data. The claim language refers to "selecting, in response to the receipt of a stream of data records from said data processor, available memory space in one of said redundancy groups to store said received stream of data records thereon." '857 Patent at Claim 6. It also refers to a "means, responsive to the receipt of a stream of data records . . . for selecting available memory space in one of said redundancy groups to store said received stream of data records thereon." Id. at Claim 11. The plain language of these claims – namely the use of the term "received . . . data records" – makes clear that the data records are already received when the selection of available memory space occurs for storage purposes.

In sum, and for all the reasons set forth above, the court adopts NetApp's construction of these claim terms, construing "in response to writing a data record to said one redundancy group"/"responsive to writing a data record to one of said redundancy groups" as "after and in reaction to the writing of a data record to a single redundancy group" and construing "in response to the receipt of a stream of data records from said data processor"/"responsive to the receipt of a stream of the data records from said data processor" as "after and in reaction to the receipt of data records from a processor."

Patent '095

This patent allows computer systems with multiple processors to accelerate performance by providing a mechanism for rapidly writing data from a processor to system memory. This system is expandable and comes with distributed shared memory, which overcomes the bottleneck of a single shared bus memory structure. The patent is directed to the use of "fast write" operations in multiprocessing systems where issues of "coherency" and stale memory data can arise. In these systems, main memory is distributed amongst multiple processors. A coherency problem is created, for example, if one processor modifies its copy of the data it is holding in its cache, but other processors have an unmodified view of the data. Microprocessors achieve coherency by: 1) informing the other processors that the copies of the data they are holding are invalid; or 2) providing an updated copy of the data to the other processors. The '095 patent claims a "fast write"

technique designed to expedite those operations that are sufficiently independent of other on-going operations that they do not implicate immediate coherency issues. Such operations can be processed "fast," meaning that they do not need to await resolution of the coherency process to be performed. By performing the coherency operation after writing the modified data and making it accessible, performance is improved by freeing processor resources more rapidly. However, the patent notes that the fast write protocol is not suitable for all of the write operations within the computer system, and the invention codes for both fast write and slow write operations. '095 Patent at 4:60-5:38.

5. "Completing [a] Write Operation Within [a] Local Processing Node" / "Completing [a] Write Operation With Respect to [a] Processor"

NetApp's Proposed Constructions	**Sun's Proposed Constructions**
Claim 1: Transferring the write data from an initiating processor to a system interface; Claims 11 & 17: Transferring the write data from an initiating processor to a system interface	Claim 1: Sun contends that the phrase need not be construed because it is clear on its face. However, if the Court construes the phrase, Sun's construction is: Data for the write operation (1) is provided to subsequent read operations **within the local processing node** to the same address as the write operation and (2) is or will be coherent within distributed shared memory; Claims 11 & 17: Sun contends that the phrase need not be construed because it is clear on its face. However, if the Court construes the phrase, Sun's construction is: data for the write operation (1) is provided to subsequent read operations **by a processor** to the same address as the write operation and (2) is or will be coherent within distributed shared memory

Here, Sun contends that the phrases need not be construed. Should the Court construe the claims, however, the parties assert that both phrases should be construed the same way, except that Sun's proposed construction of the second phrase contains one variation from its proposed construction of the first phrase, where bolded above. Thus, the disputes between the parties are whether construction is necessary to assist the jury, and if so, what is the endpoint for a write operation to be considered complete with respect to a processor or within a processing node. NetApp argues that a write operation is completed for purposes of these claims when the write data

United States District Court
For the Northern District of California

is transferred to the system interface.

As to Sun's argument that the disputed language needs no construction because the meaning of completing a write operation is clear to a person of ordinary skill in the art, while for some commonly understood words, claim construction may be resolved by judges using the application of the widely accepted meaning of those words, Phillips, 415 F.3d at 1314, an expert purporting to comprehend a non-commonly understood phrase may not impose that meaning on a term. Rather, "[b]ecause the meaning of a claim term as understood by persons of skill in the art is often not immediately apparent, and because patentees frequently use terms idiosyncratically, the court looks to those sources available to the public that show what a person of skill in the art would have understood disputed claim language to mean." Id. (internal quotations omitted). Such sources, as discussed above, include claim language, the claim specification, the prosecution history, and extrinsic evidence. This claim term is not clear on its face, and the Court turns to these sources to construe the phrase.

To begin its analysis, the court first turns to the claims themselves. These phrases appear in independent claims 1, 11, and 17 of the '095 patent. Claim 1 is for a "method for performing write operations in a multiprocessing computer system, comprising: initiating a write operation by a processor within a local processing node of said multiprocessing computer system; performing a coherency operation to at least one remote processing node in response to said write operation; completing said write operation within said local processing node prior to completion of said coherency operation if said write operation includes a specific predefined encoding; and completing said write operation within said local processing node subsequent to completion of said coherency operation if said write operation includes an encoding different than said specific predefined encoding." '095 Patent at 31:63-32:10.

As Sun notes, this claim makes it clear that write operations within the processing node can be completed prior to or subsequent to the completion of the coherency operations. The specification also recognizes this. See '095 Patent at 19:4-12; 27:17-23. The former constitutes a "fast write" operation, and the latter constitutes a "slow write" operation. The specification contains examples of slow-write operations completed subsequent to the completion of coherency activity,

United States District Court
For the Northern District of California

such as where data is transferred to other caches in the system and where no data is transferred at all. See '095 Patent at 3:44-50. NetApp's proposed construction, however, requires completion of a write operation only upon transfer of the write data to the system interface. This ignores the write invalidation procedure in which some write operations are completed without updating data in a remote cache or in the memory, and hence without transfer to a system interface.[5] See, e.g., id. at 17:57-67; 19:4-20:23 & Fig. 5. The processor need not transfer the data to the system interface in operations where updated data is transferred to the other caches or is not transferred at all, as would be required by NetApp's construction. Alpert Decl. ¶¶ 49-50. Therefore, this claim language, when read along with the specification, indicates that NetApp's construction is too narrow.

In addition, independent claim 11 reads in part: "wherein said system interface is configured to complete said write operation with respect to said processor . . ." '095 Patent at 32:41-43. This claim therefore includes an express limitation providing that the system interface completes the write operation. This limitation is not included in the other independent claims 1 or 17. This indicates that the narrow claim limitation from claim 11 should not be read into the broader claims. Uniroyal Inc. v. Rudkin-Wiley Corp., 837 F.2d 1044, 1054-55 (Fed. Cir. 1988). Because the two claims use different language, a presumption exists that the claims have different meanings. CAE Screenplates, Inc. v. Heinrich Fielder GmbH, 224 F.3d 1308, 1317 (Fed. Cir. 2000). However, while the Court cannot read into a claim a limitation that is lacking, Curtiss-Wright Flow Control Corp. v. Velan, Inc., 438 F.3d 1374, 1380-1381 (Fed. Cir. 2006), because these are two independent claims (as opposed to an independent and dependent claim), claim differentiation is characterized more generally as creating the "presumption that each claim in a patent has a different scope." Id. Nonetheless, two claims with different terminology can define the exact same subject matter. Id. Under the doctrine of claim differentiation, therefore, this Court still must look to the intrinsic evidence to define the proper claim scope, but the claim language creates a presumption that NetApp's proposed system interface limitation is not proper. See also Renishaw PLC v. Marposs Societa Per Azioni, 158 F.3d 1243, 1249 (Fed. Cir. 1998) (court is not authorized to impose a

---

[5] At the hearing, NetApp argued that even in such circumstances, a command to invalidate is sent to the system interface. However, it is unclear how an invalidation command constitutes "write data," and NetApp has not offered any persuasive evidence showing that it is.

limitation that is not required by the specification).

The specification's summary of the invention contains a general broad description of the present invention without referring to system interface. It then notes, in a subsequent paragraph, that the invention "further contemplates an apparatus for performing write operations in a multiprocessing computer system comprising a processor and a system interface." This second paragraph explains that the system interface is "configured to complete the write operation." '095 Patent at 5:58. In a third and final paragraph, the specification explains that the present invention further contemplates a computer system comprising a first processing node and a second processing node. The third paragraph also notes that the first processing node is configured to complete the write operation. Id. at 6:4-5. As Sun's expert observed, the broader first paragraph in the specification (id. at 5:39-51) generally tracks the language of claim 1. The second paragraph in the description, limited to the system interface hardware (id. at 5:52-65), generally tracks the language of claim 11. The third paragraph (id. at 5:66-6:12) tracks the language of claim 17. See also Alpert Decl. ¶ 52. The patent also confirms that the invention may be implemented in alternate forms. See, e.g., '095 Patent at 31:54-59. The specification and the plain language of the claims demonstrate that the inventor contemplated requiring the system interface architecture in only one of the three independent claims.

NetApp points to examples showing that completion with respect to a processor within a local processing node occurs when the write data is transferred to the system interface. However, these examples merely show that fast-write operations may require data transfer to a system interface to be complete. See, e.g., '095 Patent at 27:15-27, 27:43-47 (stating that fast write operations allow transfer of data to system interface before completion of coherency operations, which causes the processor to view the write operation as being complete; further describing this completion to be in local node); 31:36-41 ("[p]rocessor resources are freed upon transmission of the write operation and corresponding data to the system interface, before an appropriate coherency state is acquired by the node containing the processor"); 28:46-52 (describing completing a write operation as follows: "Since ignore signal 70 is not asserted upon the fast write transaction, the corresponding data is subsequently provide by processor 16 upon data bus 60 (shown in FIG 2).

United States District Court
For the Northern District of California

During step 322, the data is received and stored by the system interface 24. The write operation is thereby complete with respect to the initiating processor 16."); 29:45-52 (noting that "since the fast write stream transactions are completed from processors 16 by storing the transaction into SMP in queue 94 and the corresponding data within output data queue 90, processors 16 may continue with other operations while system interface 24 completes the write stream operations"). However, the patent is not limited to fast write operations. Rather, the specification teaches that fast-write operations are not always safe or proper, because in certain circumstances, coherency may be necessary sooner. See id. at Abstract. As discussed above, the claims clearly contemplate that write operations within the processing node can be completed prior to or subsequent to the completion of the coherency operations, and may be coded differently as appropriate to do one or the other.

In light of the above, NetApp's proposed construction, which requires a complete write operation to transfer write data to a system interface, is far too narrow. As to Sun's proposed alternative construction, the specification requires that coherency operations be performed either prior to or after performing the write, which provides support for the second prong of Sun's proposed constructions. Sun's construction also provides that a completed write operation requires that the written data be capable of being read, which is how one skilled in the art would define the term, given that the specification does not support NetApp's improper limitation. See Alpert Decl. ¶¶ 39-41 (complete in context of write operation requires that written data be capable of being read and this is well understood in the art) and Ex. D (citing NetApp's expert's 1987 publication). The second prong of Sun's constructions, however, are redundant of the express language of the claims, since the claims themselves already specify that one class of writes completes prior to the coherency and the other class of writes completes subsequent to coherency. Finally, Sun's "is or will be coherent" language would not be particularly helpful to a jury.

In sum, and for all the reasons set forth above, the court is deferring adopting a construction of these terms at this time. The parties shall meet and confer and propose a new joint construction in light of this claim construction order that does not import the system interface limitation and is along the lines of Sun's proposed construction, but addresses the redundancy and jury comprehension issues discussed above. The parties shall submit an agreed upon construction no later than

September 19, 2008. In doing so, the parties do not waive any rights to preserve their objections that their original proposals are correct.

<u>Patent ’630</u>

This patent is directed at an ethernet network architecture that enables ethernet networks to operate at high speed data transmission rates (multiple gigabits per second). The patent enables increased transmission rates by dividing communications into smaller parts and distributing the smaller parts among a plurality of channels. By sending the communication across multiple channels rather than a single channel, the overall data transmission rate is increased to the sum of the transmission rates for each of the multiple channels, rather than being limited to the rate for a single channel. ’630 Patent at 2:28-47. In one embodiment of the invention, the communication is divided for transmission across multiple channels at a point below the medium access control (“MAC”) layer of operation, so that the individual bytes of each frame or packet of the communication are separated and sent across each one of the channels in a round robin fashion. <u>Id</u>. at 2:36-42. When the portions of the data reach the destination computer, they are combined into a single data stream and passed to the destination computer’s MAC layer. <u>Id</u>. at 2:33-35.

6. “Portion [of a] Communication”

7. “Element [of a] Communication”

Term	**Agreed Construction**
Portion of a Communication	Part of a data stream, where the part includes “elements” of the data stream.
Element of a Communication	A constituent part of a “portion.”

The term “portion [of a] communication” appears in claims 3, 5, 12-15, 21-22, 45, 48, 50, 52-53, 73, and 76 of the ’630 patent. NetApp originally contended that this phrase should be construed as “the fraction or portion of a frame carried by one channel.” Sun argued that the term should not be limited to frames, because the frames did not include certain control data that was clearly part of the data stream. In its reply brief, NetApp changed the focus of the dispute, arguing instead that these terms had to be limited to at or below the medium access control (“MAC”) layer.

At the hearing, it became clear that the parties were unsure about the exact nature of their dispute. For example, the parties agreed that the technology only includes ethernet technology, which operates with frames, and agreed that the control data was part of the communication. The Court agrees that such conclusions are supported by the specification. Sun also conceded that frames are only found at or below the MAC layer. The parties therefore agreed to meet and confer to see if they could come to an agreement about the construction of the terms. The parties subsequently submitted the above agreed-to constructions, which the Court adopts. The Court construes "portion of a communication" as "part of the data stream, where the part includes elements of the data stream," and construes "element of a communication" as "a constituent part of a portion."

’292 Patent

The ’292 patent describes a method for maintaining consistent states of a file system, and for creating snapshots that are read-only copies of the file system. ’292 Patent Abstract. The invention uses a write anywhere file-system layout ("WAFL"). The file system progresses from one self-consistent state to another self-consistent state. The set of self-consistent blocks on disk that is rooted by a root inode is referred to as a consistency point. The root inode is stored in a file system information structure. Id. In the detailed description of the invention, the invention is described as a system that utilizes storage blocks on the disk, and disk drives provide the storage space for the file system and maintain the structure and content of the file system. The system also uses an in-memory "buffer" for holding changes to the file system prior to storing them on the disk. The data blocks are organized, described, and pointed to by inode blocks. Id. at 5-6. The inode files themselves are pointed to by a "root inode." Id. at 9:34-35. When changes to the file system occur during use, WAFL writes new or modified data to unallocated blocks on disk so that it never overwrites existing data. Id. at 12:2-4. The file system in this patent can also maintain a copy of past or old data in a consistent state by retaining it in a read-only form called a snapshot. Id. at 4:20-21.

///

///

///

United States District Court
For the Northern District of California

8. "Non-volatile Storage Means"

NetApp's Proposed Construction	Sun's Proposed Construction
A storage device that can retain information in the absence of power	Sun contends that the term non-volatile storage means should be construed as a means-plus-function limitation governed by 35 U.S.C. § 112 ¶ 6, and that the function is "storing blocks of data of a file system so that the data is not lost in the absence of power" and that the structure is "one or more disks with a block-based format (i.e., 4KB blocks that have no fragments), where the disk storage blocks are the same size as the data blocks of the file system."

To begin its analysis, the court first turns to the claims themselves. The term "non-volatile storage means" appears in the two independent claims (claims 4 and 8) of the '292 patent. Claim 4 provides a method for maintaining a file system comprising blocks of data stored in blocks of a non-volatile storage means at successive consistency points comprising the steps of:

storing a first file system information structure for a first consistency point in said non-volatile storage means, said first file system information structure comprising data describing a layout of said file system at said first consistency point of said file system;

writing blocks of data of said file system that have been modified from said first consistency point as the commencement of a second consistency point to free blocks of said non-volatile storage means;

storing in said non-volatile storage means a second file system information structure for said consistency point, said second file information structure comprising data describing a layout said file system at said second consistency point of said file system. '292 Patent at 25:11-29.

Claim 8 provides a method for creating a plurality of read-only copies of a file system stored in blocks on a non-volatile storage means, said file system comprising meta-data identifying blocks of said non-volatile storage means used by said file system, comprising the steps of:

storing meta-data for successive states of said file system in said non-volatile storage means;

making a copy of said meta-data at each of a plurality of said states of said file system;

for each of said copies of said meta-data at a respective state of said file system, marking said blocks of said non-volatile storage means identified in said meta-data as comprising a respective read-only copy of said file system. Id. at 26:1-15.

The first critical issue is whether the term is a means-plus-function limitation under 35 U.S.C. § 112, ¶ 6, which states: "An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof." In Altiris, Inc. v. Symantec Corp., 318 F.3d 1363, 1375 (Fed. Cir. 2003) (internal citations omitted), the Federal Circuit described the analysis of means-plus-function claims:

> Where a claim uses the word means to describe a limitation, we presume that the inventor used the term advisedly to invoke the statutory mandates for means-plus- function clauses. This presumption can be rebutted where the claim, in addition to the functional language, recites structure sufficient to perform the claimed function in its entirety. Once the court has concluded the claim limitation is a means-plus-function limitation, the court must first identify the function of the limitation. The court next ascertains the corresponding structure in the written description that is necessary to perform that function. Structure disclosed in the specification is corresponding structure only if the specification or prosecution history clearly links or associates that structure to the function recited in the claim.

Id. (internal citations and quotations omitted).

NetApp first contends that there is no presumption here of a means-plus-function limitation, despite the use of the word "means" in the claim term. NetApp argues that because the claims here are process claims, not apparatus claims, the presumption does not apply. See NetApp Opp. at 2 (citing O.I. Corp. v. Tekmar Co., 115 F.3d 1576, 1582-1583 (Fed. Cir. 1997)). This argument is not persuasive. O.I. Corporation does not hold that the use of "means" in a method claim falls outside of § 112 ¶ 6. See 115 F.3d at 1582-83 ("The word 'means' clearly refers to the generic description of an apparatus element, and the implementation of such a concept is obviously by structure or material. We interpret the term 'steps' to refer to the generic description of elements of a process, and the term 'acts' to refer to the implementation of such steps. This interpretation is consistent with the established correlation between means and structure. In this paragraph, structure and material go with means, acts go with steps."). NetApp cites no case so holding, and in fact, the

United States District Court
For the Northern District of California

Federal Circuit has construed "means" terms in method claims as means-plus-function limitations. See, e.g., On Demand Mach. Corp. v. Ingram Indus., 442 F.3d 1331, 1334, 1340-41 (Fed. Cir. 2006) (construing "means for" term in a patent for "system and method of manufacturing a single book copy" in a method claim as a means-plus-function limitation); J & M Corp. v. Harley-Davidson, Inc., 269 F.3d 1360, 1364 n.1 (Fed. Cir. 2001) ("Claim 17, which is a method claim, contains a nearly identical means-plus-function limitation.").

In addition, NetApp argues that reciting "means" without reciting the phrase "means for" does not give rise to the presumption of § 112 ¶ 6. However, "means for" language is not necessary to invoke the means-plus-function presumption. See, e.g., Eaton Corp. v. Rockwell Int'l Corp., 323 F.3d 1332, 1343 n.1 (Fed. Cir. 2003) (finding that "means (26 and 34) [associated with said driveline system] are written in means-plus-function format"); Unidynamics Corp. v. Automatic Prods. Int'l, 157 F.3d 1311, 1318 (Fed. Cir. 1998) ("spring means tending to keep the door closed" written in means-plus-function format); Signtech USA, Ltd. v. Vutek, Inc., 174 F.3d 1352, 1355 (Fed. Cir. 1999) ("ink delivery means" as well as some "means for" terms found to be means-plus-function elements). The means-plus-function presumption therefore applies, albeit perhaps somewhat less strongly than if the term "means" had been followed by "for." See, e.g., Greenberg v. Ethicon Endo-Surgery, 91 F.3d 1580, 1583 (Fed. Cir. 1996) ("use of the terms 'means' (particularly as used in the phrase 'means for') generally invokes section 112(6) and that the use of a different formulation generally does not"). However, the presumption is rebuttable. The Court must also examine the language surrounding the term "means" to determine whether or not a function corresponds to the term at issue and whether the claim recites sufficient structure for performing that function.

The presence of a function corresponding to the non-volatile storage means is readily apparent. Turning to the claim language, the parties do not dispute the well-understood meaning of "non-volatile" as maintained in the absence of power. The functions of the "non-volatile storage means" include storing data blocks of a file system (claims 4 and 8), storing file information structures (claim 4), storing read-only copies of a file system (claim 8) and storing meta-data for successive states of said file system (claim 8), so that the data is not lost in the absence of power.

’292 Patent at 25:11-29, 26:1-15. In light of the plain claim language, the recited function at a minimum is storing blocks of data for the file system such that information is retained in the absence of power, as both claims 4 and 8 describe this function at the outset of the claim.

NetApp argues nonetheless argues that the term “non-volatile storage means” is not explicitly linked to the function in an active way. For example, in the first limitation of claim 4, the “first file system information structure” is stored in the “non-volatile storage means,” which, according to NetApp, shows that the means is just a passive component into which something else is stored. However, storing is a function, just as “to store” is a verb, albeit a stative rather than an active verb. In other words, it is quite common to refer to something being stored in an object, rather than referring to that object’s function in a more active way. But this active/passive distinction does not undermine the conclusion that storing is a function. Nor does NetApp cite any authority that functions can only correspond to active verbs in means-plus-function claims. In addition, NetApp contradicts itself because it admits that every dictionary definition defines non-volatile storage in “functional terms,” as a storage device that retains information in the absence of power. NetApp Opp. at 5:13-15. Storing blocks of data of a file system so that the data is not lost in the absence of power, therefore, is the corresponding function. The Court now turns to the more difficult question of whether or not the claim recites sufficient structure for the means-plus-function limitation to apply.

NetApp argues that the presumption does not apply because the claim recites sufficient structure to perform the claimed function in its entirety. NetApp relies on a number of cases in support of its argument, but each is distinguishable from this case. In Greenberg v. Ethicon Endo-Surgery, 91 F.3d 1580, 1583 (Fed. Cir. 1996), the Court looked to dictionary definitions to determine whether the noun “detent” disclosed sufficient structure. While finding that the term did not call to mind a single well-defined structure, the Court held that the term, as the name for structure, had a reasonably well understood meaning in the art. However, in Greenberg, the Federal Circuit construed the term “detent mechanism,” not “detent means.” The Court specifically distinguished a case in which that patentee specifically chose to invoke § 112 ¶ 6 by invoking the “detent means” language. Id. at 1583-84. Specifically, the Court noted that “the use of

the term 'means' has come to be so closely associated with 'means-plus-function" claiming that it is fair to say that the use of the terms 'means' (particularly as used in the phrase 'means for') generally invokes section 112(6) and that the use of a different formulation generally does not." Id. at 1584. The fact that the "means" language is used here makes this case quite unlike Greenberg.

NetApp also relies on Cole v. Kimberly-Clerk Corp., 102 F.3d 524, 531 (Fed. Cir. 1996) (holding that term "perforation means . . . for tearing" was not means-plus-function limitation because the term described the structure supporting the tearing function (perforations) and went even further to describe the location of the structure and extent of that structure); Envirco Corp. v. Clestra Cleanroom, Inc., 209 F.3d 1360, 1365 (Fed. Cir. 2000) (concluding that claims recited sufficient structure, where dictionary definition of the word baffle was "a device (as a plate, wall or screen) to deflect, check, or regulate flow" so that term itself imparted structure (a surface which deflects air), and where claims described details about location and structure of baffle); and Keithley v. Homestore.com, Inc., 2007 WL 2701337, *20 (N.D. Cal. Sept. 12, 2007) (looking to dictionary definition of terms at issue, and finding that server definitions including "A computer that manages centralized data storage or network communications resources. A server provides and organizes access to these resources for other computers linked to it." disclosed inherent structure in the term, and that claim language further detailed particularized structure). Those cases, however, differ from this one in two important respects. First, the claims there detailed a more particularized structure, and usually a specific location for that structure. Such claim language is not present here. Second, the definitions in those cases outlined a more detailed structure for the term, such as a perforation, which is a structure with a series of holes designed to allow tearing, and a baffle, which is a plate, wall or screen to deflect, check, or regulate flow. The definitions for "non-volatile storage" proffered here are more general and encompass a broader range of structures.

NetApp argues that Sun's dictionary entries for "non-volatile storage" consistently define it as a storage device that retains data in the absence of power, which recites sufficient structure to perform the claim function and is the construction NetApp proposes. Furthermore, NetApp argues that certain definitions point to specific electronic components, demonstrating that structure is inherent in the term. The Microsoft Press Computer Dictionary (1991), for example, explains that

United States District Court
For the Northern District of California

"non-volatile memory" is intended to refer to "core ROM, EPROM, bubble memory, or battery backed CMOS RAM" and that the "term is occasionally used in reference to disk subsystems as well." Ganger Decl., Ex. F; see also Brandt Decl., Ex. 3 (Webster's New World Dictionary of Computer Terms, 376 (8th ed. 2000) (explaining that read-only memory is non-volatile, as are all secondary storage units such as disk drives). NetApp relies on numerous other dictionary definitions, many of which define the term "non-volatile storage" in term of its function, i.e., as a "storage device which can retain information in the absence of power." See Ganger Decl. ¶ 12 (citing IEEE Standard Dictionary of Electrical and Electronics Terms 699 (6th ed. 1996)).

Based on these definitions, it is clear that at a minimum, the term is understood as a storage system that does not lose data when its power is cut off. This ability to maintain data in the loss of power, however, is more functional than structural. Further, the myriad definitions encompass a broad array of different types of structures that may include different types of ROM or disk drives, among other things. However, the definitions do not make clear what type of structures are necessarily contemplated by the term, and the definitions themselves are highly variable. See Ganger Decl. ¶ 12 (citing various dictionary definitions which define non-volatile memory or storage to include memory chips such as read-only memory in one definition; core, ROM EPROM, bubble memory, or battery backed CMOS random access memory (RAM) and occasionally with subsystems in another definition; read-only memory and secondary storage units like disk drives in yet a separate definition; and magnetic tape, drum, or core in a different definition). NetApp's own expert includes "disks, disk arrays, flash memory drives, *and the like*" in the set of devices providing non-volatile storage. Ganger Decl. ¶ 8 (emphasis added). However, flash memory drives are very different structurally from disks, in that they are specifically designed to be small, readily detachable by users without technical expertise, and highly portable. In addition, Dr. Ganger's use of "and the like" is extremely open-ended, indicating that the set of structures corresponding to non-volatile storage are not ascertainable. While Sun's expert seems to overreach in arguing that the structure contemplated by these claims in the '292 patent filed in 1995 would be understood by one of ordinary skill in the art to include paper tape and punch cards (Brandt Decl. ¶ 65), even setting these outliers aside, the dictionary definitions and NetApp's own expert provide too broad a range of

devices that are not sufficiently similar in character to provide the corresponding structure. In addition, hard disk drives are the relevant structure disclosed in the '292 patent, yet the dictionary definitions do not focus on this structure.[6] Therefore, while NetApp argues that the term has a broad meaning in the form of many corresponding structures, citing Linear Tech. Corp. v. Impala Linear Corp., 379 F.3d 1311, 1322 (Fed. Cir. 2004) ("That the disputed term is not limited to a single structure does not disqualify it as a corresponding structure, as long as the class of structures is identifiable by a person of ordinary skill in the art."),[7] here, that class of structures is not sufficiently identifiable to defeat the means-plus-function presumption.

This case is similar to Altiris, Inc. v. Symantec Corp., 318 F.3d 1363 (Fed. Cir. 2003), in which the Court analyzed whether "means of booting" was a means-plus-function limitation, where the claim described "a means of booting said digital computer, said means of booting including a first set of commands, said first set of commands resident on said storage device of said digital computer for booting said digital computer, and a second set of commands resident on a storage device external to said digital computer for booting said digital computer." The district court concluded that this was a means-plus-function limitation because "the language referring to two sets of commands states only the location of the commands and is insufficient to define the structure that performs the function of booting." The Federal Circuit agreed. See id. at 1375-76. It noted that while "commands" represented structure (in the form of software), "commands" did not provide sufficient structure to perform the entirety of the function. Because that language was "so broad as to give little indication of the particular structure used here and [wa]s described only functionally, one must still look to the specification for an adequate understanding of the structure of that

---

[6] NetApp also argues that Sun's own expert's testimony provides additional evidence confirming that non-volatile storage refers to a structural component, such as a disk, and that Sun's expert Marshall McKusick has equated non-volatile storage with disk in 2000 and 2007 publications. See Ganger Decl., Exs. M at 12 ("non-volatile storage (i.e., disk)") and N at 128 (same)). NetApp also argues that Sun's own patent portfolio reveals many patents in which non-volatile storage is equated with a structure, generally a computer disk. Granger Decl. ¶ 16. But again, this disk limitation is in contrast to the broader dictionary definitions discussed above, further indicating that the class of structures is not sufficiently disclosed in the claims themselves to defeat the means-plus-function presumption.

[7] In addition, this cited passage in Linear pertains to identifying the corresponding structure of a claim term that was already held to be governed by § 112 ¶ 6. Id.

United States District Court
For the Northern District of California

software." The Court distinguished cases holding that the claims disclosed sufficient structure as construing patents in which "the exact structure used to accomplish the function appears in the claim language." Id. at 1376. Similarly, here, the term "non-volatile storage means" does not disclose sufficient structure to accomplish the function in the claim language, and the dictionary definitions do not disclose a clear indication of the particular class of structures used. See also Laitram Corp. v. Rexnord, Inc., 939 F.2d 1533, 1536 (Fed. Cir. 1991) ("The recitation of some structure in a means plus function element does not preclude the applicability of section 112(6). For example, in this case, the structural description in the joining means clause merely serves to further specify the function of that means. The recited structure tells only what the means-for-joining does, not what it is structurally."); Unidynamics Corp. v. Automatic Prods. Int'l, 157 F.3d 1311, 1319 (Fed. Cir. 1998) (addressing the term "spring means tending to keep the door closed" and concluding that where spring was the only recitation of structure, it did not provide sufficient structure to overcome presumption that § 112 ¶ 6 applied).

The parties both rely on cases that analyzed whether the term "storage means" was a means-plus-function limitation. None of these cases alters the analysis. NetApp notes that in Lottotron, Inc. v. Scientific Games Corp., 2003 WL 22075683 (S.D.N.Y. Sept. 8, 2003), the district court did not find a means-plus-function limitation in the term "storage means," where the dictionary definition explained that it was a "device consisting of electronic, electrostatic, electrical, hardware, or other elements into which data may be entered. . . ." Id. at *7. However, that Court did not address the myriad of different dictionary definitions that exist in this case. That case also involved a lottery wagering system – a completely different technology. In addition, neither party argued that the claim was a means-plus-function limitation. Id. Finally, other cases have held that the term "storage means" is a means-plus-function limitation. See, e.g., Apple Computer v. Burst.com, Inc. 2007 U.S. Dist. LEXIS 33863, *65-66 (N.D. Cal. May 8, 2007) (Patel, J.) (concluding that § 112 ¶ 6 applied to "storage means" where most specific dictionary definition provided that term was "any device in which information can be stored, sometimes called a memory device," noting that "description of storage as a 'memory device' underscores the conclusion that 'storage' is a functional term," as a "memory device does not connote a particular structure-such as an input

United States District Court
For the Northern District of California

port").[8]

For these reasons, the claims do not recite sufficient structure sufficient to perform the claimed function in its entirety. The term, therefore, is a means–plus-function limitation, and the Court must determine the corresponding structure. The construction of a means-plus-function limitation involves two steps. "A disclosed structure is corresponding only if the specification or the prosecution history clearly links or associates that structure to the function recited in the claim." Omega Eng'g, Inc. v. Raytek Corp., 334 F.3d 1314, 1322 (Fed. Cir. 2003). The structure must be necessary to perform the claimed function. Id.; see also Wenger Mfg., Inc. v. Coating Mach. Sys., Inc., 239 F.3d 1225, 1233 (Fed. Cir. 2001).

As discussed above, the function is "storing blocks of data for a file system so that the data is not lost in the absence of power." As to the corresponding structure disclosed in the specification, the specification discloses only one corresponding structure storing data blocks of a file system. Sun notes that the specification repeatedly and exclusively describes the invention in the context of the structures and operation of a write anywhere file-system layout (WAFL) system. See, e.g., '292 Patent at 5:49-52 ("The present invention uses a Write Anywhere File-system (WAFL)."); 5:62 ("WAFL inodes are distinct from prior art inodes."); 6:53-56 (describing WAFL inodes); 8:57-9:17 (describing WAFL directories). Sun notes that fundamental to WAFL is its use of a disk format system that is "block based (i.e., 4 KB blocks that have no fragments)." Id. at 5:48-53. The use of disks with a block based format is the only structure disclosed for storing blocks of data in a file system where the data is not lost in the absence of power. Brandt Decl. ¶ 67.

NetApp argues that Sun's definition improperly imports the characteristics of the overlying file system into the non-volatile storage, which "confuses jars and apple sauce." Ganger Decl. ¶ 18. Specifically, the requirements of 4KB blocks, blocks with no fragments, and disk storage blocks with the same size as the data blocks of the file system are not characteristics of the non-volatile

[8] The Court in Apple, however, also construed the term "random access storage means" and found that it was not a means-plus-function limitation. In so finding, the Court noted that both experts essentially conceded that the functional language "storing the time compressed representation" was specific enough to connote a particular class of structures, id. at *67. By contrast, this issue is hotly contested here. Moreover, RAM storage appears to be at least slightly narrower than the "non-volatile" storage at issue here. In addition, the Federal Circuit cases discussed above require more specificity in the type of structure disclosed.

United States District Court
For the Northern District of California

storage device. Rather, these are characteristics of the file system, as is evidenced by the fact that Sun draws its proposed structure from a section of the specification describing the file system layout. Ganger Decl. ¶¶ 18-19; '292 Patent at 5. This part of the specification does not focus on the non-volatile storage device, but rather on how the overlying file system organizes the raw capacity. See, e.g., '292 Patent at 5:49-52 (describing inodes, files, and directories, but not non-volatile storage). And, indeed, block fragments are a file system concept. Ganger Decl. ¶¶ 19-20. At the hearing, Dr. Ganger further explained that disks do not understand fragments, and that storage devices read and write data, which they understand in units of KB. Sun's expert Dr. Brandt agreed, but noted that when WAFL uses storage blocks, it accesses that data in 4 KB blocks. Dr. Ganger conceded that while not all disks in this invention had to store data in multiples of 4 KB, it would be far less convenient if the disks were not divisible by 4 KB for the purposes of this invention.

In sum, in looking at the specification in combination with the experts' explanation of the invention, the structure of the file system is not the same as the structure of the non-volatile storage, so Sun's proposed file system limitations are not necessary structure to perform the claimed storage function. Specifically, fragments are a file system concept and should not be imported into the structure of the non-volatile storage. And while the disks store 4 KB blocks, it is not clear from the specification that the disks themselves must have a 4KB block structure. As Dr. Ganger notes, the patent specification repeatedly explains that the thing in which the various file system structures are stored is a disk, and the corresponding structure, therefore, must be disks and disk arrays.[9] See Ganger Decl. ¶ 17; see, e.g., '292 Patent at 9:48-49 ("written to disk"); 12:2-3 (WAFL always writes new data to unallocated blocks on disk). It is also clear that for purposes of this invention, such disks and disk arrays must be suitable for storing data in 4 KB blocks, since in order for WAFL to be a disk format system that is block based (i.e. 4 KB blocks that have no fragments), the disk must be formatted such that it can readily store 4 KB blocks. Supp. Brandt Decl. ¶ 7; '292 Patent at 5:48-53.

In sum, and for the reasons stated above, the Court adopts the following construction of this term, which it construes as a means-plus-function limitation governed by 35 U.S.C. § 112, ¶ 6.

---

[9] At the hearing, Dr. Brandt did not contest inclusion of disk arrays, and noted that the multiple disks could be used.

Function: storing blocks of data of a file system so that the data is not lost in the absence of power; Structure: disk or disk arrays that are suitable for storing 4 KB blocks.

9. "Meta-Data for Successive States of Said File System"

NetApp's Proposed Construction	**Sun's Proposed Construction**
Information that describes "successive states of a file system" (as construed herein)	A block map file for recording snapshots of the file system

The parties dispute whether "successive states of a file system" refers only to snapshots and whether "meta-data" in this context must be a block map file. The term "meta-data for successive states of said file system" appears in claim 8 of the '292 patent, which is set forth above, and pertains to creating a plurality of read-only copies of a file system stored in blocks of a non-volatile storage means. Both parties agree that such read-only copies are snapshots, and that this claim pertains to creating snapshots. Claim 8 recites multiple steps for creating snapshots and recording these snapshots as a copy of meta-data. Specifically, the claim requires that the meta–data first be stored for successive states of said file system, then be copied "at each of a plurality of said states of said file system," then be marked. '292 Patent at 26:1-15. At the end of this process, a snapshot has been created. However, Sun's construction renders this process circular and collapses the separate steps. Under Sun's construction, the meta-data that is stored in the first step of this process would be the snapshot – more particularly, a file for recording the snapshots. However, the meta-data itself cannot be the snapshot, because the meta-data is what the invention uses to make the snapshot.

In addition, the claim language shows that the snapshots are only a subset of the meta-data for successive states that are stored in claim 8. The plain language of the claim notes that after the meta-data is stored for successive states of the said file system, copies of that meta-data are made "at each of a plurality of said states of said file system." While "plurality" means that copies are made at more than one of the successive states, it does not require that copies be made at every successive state. In other words, the plain meaning of plurality does not require that copies be made at *all* successive states. This, in turn, means that not all meta-data is marked. The inventor knew how to refer to the entirety of a set, but chose not to do so. For example, in the same claim, the inventor

United States District Court
For the Northern District of California

required that "each of said copies of said meta-data" be marked. Thus, all copies at this step are marked. However, at the earlier copying stage, the inventor chose not to use this all-inclusive language, and instead opted to use the phrase "at each of a plurality." This claim, therefore, does not require that all meta-data in these claims be stored as snapshots.

As to whether or not meta-data in this claim must be in the form of a block map file, NetApp argues that the doctrine of claim differentiation mandates rejection of Sun's proposed construction. Specifically, dependent claims 9 and 10 add the limitations "means for recording multiple usage bits per block" and "blockmap comprising multiple bit entries for each block," respectively. NetApp argues that because the "blockmap" limitation is the only difference between claims 9 and 10, that limitation may not be read into independent claim 8. NetApp Opp. at 15; Ganger Decl. ¶¶ 37-38. "In the most specific sense, 'claim differentiation' refers to the presumption that an independent claim should not be construed as requiring a limitation added by a dependent claim." Curtiss-Wright Flow Control Corp. v. Velan, Inc., 438 F.3d 1374, 1380 (Fed. Cir. 2006). While claims 9 and 10 contain additional language that is more specific than just requiring a block map alone, and thus Sun's construction would not render these dependent claims entirely superfluous, these dependent claims indicate that the block map requirement should not be imported into the independent claim.

In addition, the narrowing limitations of dependent claims 11-13 and 18-19 might be excluded under Sun's definition, because the proper construction for "meta-data" must be at least broad enough to include different types of meta-data such as pointers to a hierarchical tree of blocks, structures representing files of the file system, and a root structure referencing structures representing files of said file system. See, e.g., '292 Patent at Claim 11 ("wherein said meta-data comprises pointers to a hierarchical tree of blocks"); Claim 12 ("wherein said meta-data comprises structures representing files of said file system"); Claim 18 ("wherein said meta-data comprises a root structure referencing structures"). While Sun argues that the requirement of a block map file for recording snapshots of the file system in the independent claim does not exclude the presence of the additional structures listed in the dependent claim terms, see Brandt Supp. Decl. ¶ 24, the use of the "comprising" language in the dependent terms indicates that the meta-data in these dependent terms must include the pointers, structures, etc. referred to in those dependent claims. Therefore, the

claim language indicates that the blockmap limitation should not be included in this construction.

Turning to the specification, Sun seems to blur the distinction between what meta-data is and where it is stored. Data is stored in a file. Meta-data may be stored in a block map file. The specification makes clear that meta-data does not comprise the files themselves. See '292 patent at 5:52-56 (WAFL uses files to store meta-data that describes layout of file system); 9:21-24 (meta-data is kept in files). The blockmap file indicates which disk blocks are allocated. Id. at 5:57-58. The block map file indicates whether a block is part of the active file system or a snapshot by storing either a one or a zero in the bit position in the block map file. Id. at 10:7-15; 9:66-10:1; 4:39-43. These files do not define what meta-data is. Therefore, Sun's proposed construction, insofar as it equates meta-data with block map file, is incorrect.

Sun relies on the "Summary of the Invention" section, which states:

> The present invention prevents new data written to the active file system from overwriting "old" data that is part of a snapshot(s). It is necessary that old data not be overwritten as long as it is part of a snapshot. This is accomplished by using a multi-bit free-block map. Most prior art file systems use a free block map having a single bit per block to indicate whether or not a block is allocated. The present invention uses a block map having 32-bit entries.

'292 Patent at 4:33-40. As Sun notes, these unambiguous statements, which refer to the invention and not just a single embodiment, make clear that a block map is a necessary feature of the invention. However, this summary does not discuss meta-data at all and does not state that it is important that the meta-data be a block map file. Rather, it merely states that the present invention's use of a block map is key. This description does not weigh in favor of incorporating these limitations into the term at hand.

In addition, as NetApp argues, even in the preferred embodiment, the term at issue is not limited to a block map file, because the meta-data that is copied to make a new snapshot is a root inode, not a block map file. '292 Patent at 18:13-16; 18:65-19:6. In this preferred embodiment, the process of creating a snapshot includes a snapshot inode that is similar to the root inode and is stored in a fixed location inside the inode file. In addition, a copy is stored in the block map file.

In conclusion, Sun's proposed construction is not proper. At the same time, NetApp's proposed construction simply repeats much of the language in the disputed claim term, which is not

particularly helpful to the jury. The specification explains that WAFL keeps information that describes a file system in files known as meta-data. ’292 Patent at 9:19-20. A snapshot is a read-only copy of an entire file system at the particular instant when the snapshot is created. Id. at 17:66-18:1. Claim 8 describes a method for creating snapshots. Therefore, the Court proposes construing “meta-data for successive states of said file system” as “information describing a copy of the structure of the active file system (such as a copy of a root inode) at a series of successive points in time.” The parties shall meet and confer about the propriety of this proposed construction in light of this claim construction order. The parties shall notify the Court as to whether they agree to this construction, or submit a different agreed upon construction that comports with the guidance in this order no later than **September 19, 2008.** In doing so, the parties do not waive any rights to preserve their objections that their original proposed constructions were correct.

10. “File System Information Structure”

NetApp’s Proposed Construction	Sun’s Proposed Construction
A data structure containing information about the layout of a file system	Data structure that contains the root inode of a file system in a fixed location on disk

Here, the parties dispute whether or not the file system information structure contains a root inode, and whether the data structure is in a fixed location on disk.

The term file system information structure appears in asserted claim 4, as well as non-asserted claims 2, 5, and 7 of the ’292 patent. NetApp’s proposed construction is “a data structure containing information about the layout of a file system.” If the Court were to replace the term with this proposed language, however, claim 4 would refer to “storing a first ‘data structure containing information about the layout of a file system’ for a first consistency point in said non-volatile storage means, said first data structure containing information about the layout of a file system’ comprising data describing a layout of said file system at said first consistency point of said file system.” NetApp’s construction thus improperly repeats other language within the claim and ascribes meaning to the term that is already implicit (indeed, explicit) in the remainder of the claim. See, e.g., Mangosoft, Inc. v. Oracle Corp., 525 F.3d 1327, 1330-1331 (Fed. Cir. 2008) (refusing to

United States District Court
For the Northern District of California

adopt proposed construction that ascribed no meaning to claim term that was "not already implicit in the rest of the claim"). NetApp's construction, therefore, would create a strange and awkward redundancy in the claim language.

NetApp argues that Sun's construction requiring a fixed location reads a limitation from dependent claim 6 into independent claim 4, because claim 6 requires that the non-volatile storage means "comprise fixed predetermined locations" of said means. However, this dependent claim does not require that the single file information structure recited in claim 4 be fixed, so the doctrine of claim differentiation is inapplicable. In addition, dependent claims 5 and 6 contain additional limitations, such as overwriting copies of file system information structure, so Sun's interpretation would not render the dependent claims superfluous.[10] As a whole, therefore, the claim language indicates that NetApp's proposed construction is redundant, and does not weigh against Sun's fixed location limitation. However, the Court must turn to the specification in order to construe the meaning of the term.

As a preliminary matter, the specification seems to equate the file system information structure with the "fsinfo" by placing "fsinfo" next to or within the phrase "file information structure." For example, the specification twice refers to the "file system information (fsinfo) structure" and twice refers to the "file system information (fsinfo) block." '292 Patent at 9:33-36, 10:57-65, 13:63-64. The specification also equates the terms file information structure and block by using the terms interchangeably. For example, the specification notes that Figure 15 illustrates a fsinfo structure 1510, but then Figure 15 labels structure 1510 "fsinfo block." Id. at Figure 15 and 10:57-58.

The specification states that the root inode is kept in a fixed location on disk referred to as the fsinfo block. '292 Patent at 9:33-35. See also id. at 10:57-64. Based on this language of the specification, Sun argues that a person of ordinary skill in the art would recognize that file system information structure must include the root inode and must be located at a fixed location. See Brandt Decl. ¶ 83. According to Sun, the fixed, known location of the fsinfo block is necessary to

---

[10] Insofar as NetApp argues that the fact that claim 19 expressly recites a root inode supports its argument, corresponding independent claim 8 does not contain the term "file system information structure," so these claims are not particularly helpful to its construction.

United States District Court
For the Northern District of California

claim 4. Brandt Decl. ¶ 84. For example, the specification teaches the file system changes consistency points after the root inode is written to disk. ’292 Patent at 4:11-18, 12:16-20; see also Brandt Decl. ¶ 83. Therefore, according to Sun’s expert, the file system information structure cannot operate to change consistency points absent the presence of the root inode in that structure. Brandt Dec. ¶ 83. Because recovery of the file system is only possible if the root inode can be located by the file system, and the specification teaches that the file system must be located at a fixed location, it appears that the structure must include a root inode and be at a fixed location. See ’292 Patent at 9:27-35 (noting that the inode file may be written anywhere on disk, unlike the prior art that writes inode tables to a fixed location on disk; the inode file is pointed to by a root inode that is kept in a fixed location on disk referred to as the file system information block.); 10:57-65 (root inode of file system is kept in a fixed location on disk so that it can be located during booting of the file system); 11:3-5 (noting that two identical copies of the fsinfo structure are kept in fixed locations on disk). Thus, according to the specification, the root inode is in a fixed location on disk**,** which is referred to as the file system information block. The specification discloses this in definitional language, and this requirement does not appear to be confined to the preferred embodiment only.

NetApp notes that the specification also describes the file system information structure as one that “also contains information including the number of blocks in the file system, the creation time of the file system, etc. . . . [and] a check-sum.” ’292 Patent at 10:65-11:1. While this is true, this portion of the specification seems to include such data as optional, while the root inode must be included in the fsinfo structure. See above; see also ’292 Patent at 10:57-11:3 (noting that root inode is kept in fixed location on disk, and that except for the root inode, the miscellaneous information such as number of blocks in the file system, creation time of the file system, etc., can be kept in a meta-data file in an alternative embodiment).

However, while the specification indicates that the root inode is kept in a fixed location on disk – the file system information block – the specification expressly teaches that the file system need not necessarily use a root inode as the data structure when copying *snapshots*: “Snapshots are created by duplicating the root data structure of the file system. In the preferred embodiment, the

root data structure is the root inode. However, any data structure representative of an entire file system could be used." Id. at 18:13-16. NetApp therefore argues that any data structure representative of an entire file system could be stored in the fsinfo block in lieu of a root inode. Ganger Decl. ¶ 48. However, this portion of the specification does not contain an explicit reference to the file system information structure. Rather, this portion of the specification addresses snapshots and describes "snapshot inodes" as only "similar" to root inodes, not identical. '292 Patent at 18:12. By contrast, the specification's discussion of the file system information structure addressed above is contained in the portion of the specification addressing the WAFL layout, which is fundamental to the invention: "The present invention uses a Write Anywhere File-system Layout (WAFL)," id. at 5:48-49, although some "specific details" pertaining to disks might be varied. Id. at 5:40-43. Therefore, even though the snapshot part of the specification discusses copying a root inode to form a snapshot and then alludes to a possible alternative data structure that could be used to make a snapshot, it does not state that the file system information block or structure can do without a root inode. The abstract also discusses a file system that progresses from one self-consistent state to another self-consistent state, in which the blocks on disk are rooted by a root inode. It further notes that the method also creates snapshots, but does not limit the snapshot creation process to a root inode data structure in the general discussion. In other words, the discussion of the file system and snapshot creation are related, but separate.

In sum, the claim language and specification indicate that the "file system information structure" contains a root inode and is on a fixed location on disk. The Court, therefore, adopts Sun's construction of this term, and construes it as "data structure that contains the root inode of a file system in a fixed location on disk."

’211 Patent

This patent is a continuation of the '292 patent and is also directed to a method for keeping a file system in a consistent state. It shares the same specification as the '292 patent. The file system here utilizes a root inode which contains pointers that directly point to a special meta-data file, the inode file, that contains the inodes of all of the other files in the file system. '211 Patent at 9:25-33.

///

11. "Root Inode"

NetApp's Proposed Construction	Sun's Proposed Construction
An inode that points directly and/or indirectly to all the blocks in a consistent state of a 'file system' (as construed herein)	The index node data structure stored in a fixed location that roots a set of self-consistent blocks on the storage system that comprise the file system

The phrase "root inode" appears in independent claims 1, 9, and 17 of the '211 patent. These claims have similar limitations. Claim 9 is for: "a device comprising:

a processor;

a memory; and

a storage system including one or more hard disks;

wherein said memory and said storage system store a file system; and

wherein said memory also stores information including instructions executable by said processor to maintain said file system, the instructions including steps of (a) maintaining an on-disk **root inode** on said storage system, said on-disk **root inode** pointing directly and indirectly to a first set of blocks on said storage system that store a first consistent state of said file system, and (b) maintaining an incore **root inode** in said memory, said incore root inode pointing directly and indirectly to buffers in said memory and a second set of blocks on said storage system, said buffers and said second set of blocks storing data and meta-data for a second consistent state of said file system, said second set of blocks including at least some blocks in said first set of blocks, with changes between said first consistent state and said second consistent state being stored in said buffers and in ones of said second set of blocks not pointed to by said on-disk inode." '211 Patent at 24:39-62 (emphasis added).

The fundamental difference between the parties' constructions is whether the root inode is required to reside in a fixed location. NetApp's proposed construction, which includes pointing directly and/or indirectly to all the blocks" is somewhat redundant, as the claim language itself describes which sets of blocks or buffers the on-disk or incore root inodes point to. However, since the claim language describes more specific "pointing," it does not render NetApp's claim language

completely superfluous, and the Court turns to the specification for guidance.

Multiple portions of the patent specification teach that the root inode roots a set of blocks that comprise the file system in a consistent state. See, e.g., ’211 Patent at 11:20-27, 11:65-67. The specification teaches that the “root inode 1510B of a file system is kept in a fixed location on disk so that it can be located during booting of the file system.” Id. at 10:59-61; see also id. at 9:33-35 (root inode is kept on a fixed location on disk referred to as the file system information block described below). The root inode is the inode that locates the inode file, which stores inodes for the other files in the file system. Id. at 9:32-33. The invention teaches that the on-disk root inode is stored in a fixed location on the fsinfo block to enable the location of the inode files, which are written to any available locations on disk and may move around, in accordance with the “write anywhere” nature of the file system. Id. at 9:19-24. See also Brandt Decl. ¶¶ 127-29. The abstract itself notes that the “set of self-consistent blocks on disk is rooted by a root inode.” The use of the terms “root” and “rooted” suggest that the root inode is not rootless but rather is fixed or stored at some set location, although their primary meaning is serving as the base of the tree. ’211 Patent at 11:24-25. These portions of the specification address the on-disk root inode.

The claims reveal that there is also another type of root inode – the incore root inode. See, e.g., Claim 1 of ’211 Patent. The parties agree that “incore” means “in memory” as opposed to on-disk. Incore memory contains contents not written to disk and not stored in a fixed location. See ‘211 Patent at Fig. 8 (showing an incore inode that includes incore information, a copy of the on-disk inode, pointers, etc.). An incore root inode, like other information temporarily stored in memory, is often kept in whatever location in memory is allocated by the memory management system, not in a fixed location. See Ganger Decl. ¶¶ 75-76. Sun concedes that the incore root inode is stored in a buffer structure that is not located in a fixed location in memory and includes structures absent from the on-disk root inode. See Sun Reply at 22-23. Sun, however, argues that while a temporary copy of a root inode is stored in memory, the corresponding root inode is always stored in a fixed position on the disk, so the incore copy is really still in a fixed location. However, the fact that the incore’s original predecessor exists on a fixed location does not mean that the incore copy is on a fixed location. In any event, Sun’s construction does not make this point clear. Consequently,

United States District Court
For the Northern District of California

Sun's construction improperly limits the root inode to a fixed location, because it does not adequately distinguish the incore copy which need not be in a fixed location on a disk. However, Sun's construction would be remedied by adding that the in-memory copy of the root inode (the incore copy) is not stored in a fixed position on disk.

NetApp also argues that even an on-disk root inode is not necessarily stored at a fixed location. NetApp largely relies on Dr. Ganger's declaration, but Dr. Ganger does not point to the specification in making his observations, so this extrinsic evidence is not particularly persuasive. See Ganger Decl. ¶¶ 79-80. Dr. Ganger notes that there are other mechanisms for ensuring that the root inode can be located. For example, the file system could store a pointer to the root inode in a fixed location. But a pointer that points to other inodes in non-fixed locations is really just another way of describing a root inode. Dr. Ganger also notes that a file system could have a set of predetermined locations that might hold the root inode. However, Dr. Ganger has not demonstrated how the specification might contemplate such an embodiment, and the specification itself repeatedly refers to a "fixed location," as discussed above. However, if the set were small enough that the root inode could be readily located, it might constitute a fixed location (or at least its equivalent), as Sun's expert acknowledged at the hearing.

NetApp also argues that the specification shows that at the precise moment a new snapshot is created, at least one copy of the root inode for the active file system is not stored in a fixed location. However, these portions of the specification are describing snapshot inodes, not root inodes. See, e.g., '211 patent at 17:6-63, 18:1-10, 18:17-19, 18:57-67, 19:5-8. This copy of the root inode is referred to as a snapshot inode, which is created by duplicating the root data structure of the file system. Id. at 18:1-15. This copy, therefore, is not the root inode itself. To the extent that the specification, in discussing snapshot creation, notes that another data structure representative of the entire file system could be used in lieu of a root inode, the term being defined here is "root inode," not "any data structure representative of an entire file system." See id. at 18:7-10.

As to the language in Sun's construction that the root inode "roots a set of self-consistent blocks on the storage system that comprise the file system," the summary of the invention teaches that the "file system progresses from one self-consistent state to another self-consistent state. The

United States District Court
For the Northern District of California

set of self-consistent blocks on disk that is rooted by the root inode is referred to as a consistency point (CP)." Id. at 4:15-18; 11:64-65 (file system progresses from one self-consistent state to another self-consistent state, and set of self-consistent block on disks is rooted by root inode 1510B). The specification, therefore, supports this aspect of Sun's construction.

To the extent that NetApp relies on nCube Corp. v. SeaChange Int'l Inc., 436 F.3d 1317, 1322 (Fed. Cir. 2006), for the proposition that it is inappropriate to read a limitation into a claim that is "wholly apart from any need to interpret what the patentee meant by particular words or phrases in that claim," that case is inapposite. There, the Federal Circuit rejected a construction that imported a limitation from a dependent claim into an independent claim. Here, Sun's construction does not read in limitations from a dependent claim, but describes the scope of the invention set forth in the written description.

To the extent that NetApp incorporates its "and/or" construction of the term "directly and indirectly" (discussed below), and the "file system" (discussed above), these aspects of NetApp's construction fail for the same reasons.

In sum, and for the reasons set forth above, the Court adopts Sun's construction as modified below, and finds that the term "on-disk root inode" means "the index node data structure stored in a fixed location on disk that roots a set of self-consistent blocks on the storage system that comprise the file system" and that the term "incore root inode" means "the copy of the on-disk root inode that is contained in memory; this copy need not be in a fixed location in memory."

12. "Pointing Directly and Indirectly to Buffers in Said Memory and a Second Set of Blocks on Said Storage System"

NetApp's Proposed Construction	Sun's Proposed Construction
Pointing directly to blocks and/or buffers, and/or indirectly to blocks and/or buffers	Pointing directly and indirectly to buffers in said memory and pointing directly and indirectly to a second set of blocks on said storage system

The main dispute between the parties is whether or not this term requires both direct and indirect pointing to both blocks and buffers, or can be satisfied by either indirect or direct pointing to

United States District Court
For the Northern District of California

either blocks or buffers. The phrase appears in independent claims 1, 9, and 17 of the ’211 patent. These claims have similar limitations. Again, Claim 9 is for: “a device comprising:

a processor;

a memory; and

a storage system including one or more hard disks;

wherein said memory and said storage system store a file system; and

wherein said memory also stores information including instructions executable by said processor to maintain said file system, the instructions including steps of (a) maintaining an on-disk root inode on said storage system, said on-disk root inode pointing directly and indirectly to a first set of blocks on said storage system that store a first consistent state of said file system, and (b) maintaining an incore root inode in said memory, said incore root inode **pointing directly and indirectly to buffers in said memory and a second set of blocks on said storage system**, said buffers and said second set of blocks storing data and meta-data for a second consistent state of said file system, said second set of blocks including at least some blocks in said first set of blocks, with changes between said first consistent state and said second consistent state being stored in said buffers and in ones of said second set of blocks not pointed to by said on-disk inode.” ’211 Patent at 24:39-62 (emphasis added).

A plain reading of the claim language, “root inode pointing directly and indirectly to buffers in said memory and a second set of blocks on said storage system,” supports Sun’s construction. The use of “and” rather than “or” indicates that the root inode points both directly to some buffers and indirectly to some other buffers, and it also requires that the inode point directly to some blocks and indirectly to other blocks. As the Federal Circuit has explained, “[a] claim must be read in accordance with the precepts of English grammar.” See In re Hyatt, 708 F.2d 712, 714 (Fed. Cir. 1983). The inventor chose to use the language “and,” which has a conjunctive meaning, rather than “or.” NetApp’s proposed construction, however, replaces “and” with “and/or,” which disregards the plain claim language. Under NetApp’s construction, any one of at least eight different alternatives would satisfy the claim language, which ignores the inventor’s use of the conjunctive rather than disjunctive.

NetApp argues that the language surrounding the disputed term supports its position that there is no requirement for both direct and indirect pointers to both buffers and blocks. NetApp notes that there is a parallel structure in the claim in which the first set of blocks (limitation a) is the group that represents the first consistent state of the file system, and the second set of blocks together with buffers (limitation b) is the group that represents the second consistent state of the file system. NetApp argues that each of these groups is to be treated as a whole. But since the disputed term only appears in the second group, this structure instead indicates that the "and" used to link the "buffers and second set of blocks" describes those things as a "single combined group." In contrast, NetApp's use of "and/or" would break up this single combined group into two parts. Nor does Sun's construction apply the adverb "directly and indirectly" separately to the blocks and buffers, as NetApp argues. While Sun separates out the requirements in this way for clarity's sake (directly and indirectly to blocks and directly and indirectly to buffers), Sun's construction merely means that the term applies "directly and indirectly" to both "blocks and buffers," whereas NetApp's construction does not.

At the hearing, NetApp provided some examples of colloquial speech in which "and" is used disjunctively, for example: "the stars and stripes are red, white and blue," "we were eating and drinking milk and cookies," and "children and adults were drinking milk and juice." However, these examples are not persuasive for two reasons. First, these examples are of casual speech, not the formal written language of patents, where the inventors choose their words in a much more deliberate manner. In addition, these examples are not parallel to the term at issue, because they combine subject nouns and their transitive verbs with their direct noun objects or adjectives, whereas the disputed term combines subject nouns and verbs with adverbs and prepositional objects.

NetApp's reliance on Ortho-McNeil Pharm., Inc. v. Mylan Labs., Inc., 520 F.3d 1358, 1362 (Fed. Cir. 2008) is misplaced, as the claim language here is quite dissimilar. There, the Court noted that "[t]he claim also does not use and in isolation but in a larger context that clarifies its meaning. Specifically, *and* appears in conjunction with the adverbs *independently* and *together*. As the district court explained, these terms signal that *and* links alternatives that occur under the different conditions of independence or togetherness." Id. (construing *and* in claim that stated "R2, R3, R4,

and R5 are independently hydrogen or lower alkyl *and* R2 and R3 and/or R4 and R5 together may be a group of formula") (emphasis in original). The Court found that it would not make sense to require two mutually exclusive subsets of compounds, one in which the compounds were independent and one in which the same compounds were not independent and constituted a group. Here, there is no similar use of the phrase "independently and together." In addition, unlike here, the Court in Ortho-McNeil found that construing "and" in the conjunctive would have rendered dependent claims meaningless. Id.

The claim language, therefore, supports Sun's construction. Turning to the specification, the specification teaches that inodes may point to "indirect and/or direct blocks." '211 Patent at 7:55-57. This part of the specification, however, is not referring to the root inode, which plays the special role of rooting the entire consistent file system. In fact, the inventor's use of "and/or" in the specification with regard to non-root inodes undermines NetApp's argument that the word "and" in the claims has the same meaning as "and/or," as the inventor clearly knew how to use that broader language.

Generally, the specification makes clear that the buffers and blocks to which the incore root inode points serve a specific purpose: to store data and meta-data for a second consistent state of the file system. '211 Patent at 24:4-6, 55-57, 25:37-38. The file system maintains at least three bookkeeping files in the form of meta-data files, to keep data in a consistent state: the inode file, the block map file, and the inode map file. '211 Patent at 9:19-24; Brandt Decl. ¶ 105. The inode file contains inodes describing all other files in the file system, including the block map and inode map files. '211 Patent at 9:26-27. The inode file is pointed to by an inode referred to as the root inode. Id. at 9:32-33. The block map file keeps track of which blocks are allocated in the current file system, which blocks are used by snapshots, and which blocks are free for the file system to write data onto. Id. at 9:50-65. The inode map file is used by the file system to keep track of which inodes in the inode file are unallocated. Id. at 10:20-49.

The specification teaches that the root inode points to the inode file directly and to the other meta-data files and data files indirectly. Id. at 11:20-27 (describing root inode that describes inode file that in turn describes the rest of the files in the file system including meta-data files, so that the

United States District Court
For the Northern District of California

root inode is viewed as the root of the tree of blocks). Thus, the root inode points directly to the inode, and indirectly to the other data files in order to serve the claimed function of rooting "buffers and said second set of blocks storing data and meta-data for a second consistent state of said file system." Brandt Decl. ¶ 111. The incore root inode, on the other hand, points to the inode file buffer holding inodes for modified files and the block map file, and also points indirectly to buffers holding modified data for these files. '211 Patent at 16:4-17:57 & Fig. 17L. Once these buffers are sent to disk, the incore root inode directly and indirectly points to a new set of blocks comprising the changed data and inodes (id. at 17:35-37) as well as directly to the unchanged blocks of the inode file and indirectly to the inode's corresponding unchanged data blocks. Thus, according to Sun's expert, the specification teaches that the incore root inode points directly and indirectly to buffers and directly and indirectly to a second set of blocks on the said storage system. Brandt Decl. ¶ 112-120.

While NetApp argues that there is no requirement that the incore root inode have both direct and indirect pointers, and that, for example, a file system can be in a state where the incore root inode points to all buffers directly, NetApp relies heavily on extrinsic expert opinion and does not point to any clear language in the specification supporting its arguments. Rather, Dr. Ganger notes that Figures 8 and 3 show that if all 16 blocks to which the incore root inode points directly are modified, then the 16 buffer pointers will point directly to the contents of the second consistent state, but that no block pointers will do so. Thus, he concludes, if all 16 blocks of the root inode of the first consistent state change, the incore root inode for the second consistent state will not point directly to any blocks in the second consistent state. Ganger Decl. ¶ 64. However, Dr. Brandt contests this argument, maintaining that the root inode includes a copy of the on-disk inode, which points to the blocks comprising the first consistent state even when all such blocks have undergone changes and are represented by buffers. Supp. Brandt Decl. ¶ 41 (citing '211 patent at 7:58-60 and Fig. 10 Element 1010D).

Dr. Ganger also argues that there would not be any indirect pointers to buffers by the root inode group in a situation in which the only changes between the first consistent state and second consistent state were to inodes. Ganger Decl. ¶ 65. However, while Dr. Ganger refers to the

United States District Court
For the Northern District of California

specification's description of the general file system, he does not point to any support in the specification for his inferences on which he bases his conclusion. In addition, Dr. Brandt contests this conclusion on the ground that Ganger incorrectly assumes that the incore inodes are the same as on-disk inodes, among other things. Supp. Brandt Decl. ¶ 42. In sum, the experts contest whether the specification discloses systems in which the incore root inode may not have both direct and indirect pointers. However, NetApp's expert Dr. Ganger has not pointed to any persuasive intrinsic evidence in support, so his extrinsic evidence does not counsel interpreting this claim term in a manner inconsistent with the clear claim language.

In sum, and for the reasons set forth above, the Court adopts Sun's construction and finds that the term means "pointing directly and indirectly to buffers in said memory and pointing directly and indirectly to a second set of blocks on said storage system."

13. "Consistent State"/"State of a File System"

NetApp's Proposed Construction	**Sun's Proposed Construction**
A set of blocks on disk, rooted by a 'root inode' (as construed herein), that includes all the blocks required for the data and file structure of a 'file system' (as construed herein)	A set of storage blocks for that file system that includes all blocks required for the data and [file] structure of the file system

The parties originally appeared to dispute whether it was proper to add the word "file" before the phrase "structure of the file system." However, Sun has now agreed to add this language to its proposed construction, so this is no longer at issue. The dispute now boils down to whether or not the set of blocks needs to be rooted by a root inode. To begin its analysis, the court first turns to the claims themselves. The term "consistent state" appears in the '211 patent, while the term "state of a file system" appears in the '292 patent. The parties agree that the construction of these two claim terms should be the same. The two patents share the same specification. The terms appear in claims 1-3, 9-11, and 17-19 of the '211 patent and claim 8 of the '292 patent.

Turning to the claim language, NetApp's proposed construction requiring that the consistent state be rooted by a root inode is quite repetitive of the language already in the claims. See Mangosoft, Inc. v. Oracle Corp., 525 F.3d 1327, 1330-1331 (Fed. Cir. 2008) (refusing to adopt

United States District Court
For the Northern District of California

proposed construction that ascribed no meaning to claim term that was "not already implicit in the rest of the claim"). For example, Claim 1 of the '211 patent refers to an "on-disk root inode pointing directly and indirectly to a first set of blocks on said storage system that store a first consistent state of said file system." Claim 9 refers to an "on-disk root inode pointing directly and indirectly to a first set of blocks on said storage system that store a first consistent state of said file system." In other words, the claims themselves already describe the root inode's relationship to the consistent states of the file system. NetApp's inclusion of the phrase "rooted by a root inode (as construed herein)" in its proposed construction of "consistent state" creates a confusing repetition, because "root inode" is a separate claim limitation. Specifically, if one inserts NetApp's construction of "consistent state" into claim 1, for example, the following repetitive statement results: "maintaining an on-disk root inode on said storage system, said on-disk root inode pointing directly and indirectly to a first set of blocks on said storage system that store a first [set of blocks on disk, rooted by a root inode, that includes all the blocks required for the data and file structure of a file system] of said file system." '211 Patent at 23:64-67 (bracketing NetApp's proposed construction). If one were to also insert NetApp's proposed construction of "root inode" into the claim, an even more confusing tautology results, as NetApp's construction of "root inode" itself refers to consistent states: "maintaining an on-disk [inode that points directly and/or indirectly to all the blocks in a consistent state of a file system] on said storage system, said on-disk [inode that points directly and/or indirectly to all the blocks in a consistent state of a file system] pointing directly and indirectly to a first set of blocks on said storage system that store a first [set of blocks on disk, rooted by an inode that points directly and/or indirectly to all the blocks in a consistent state of a file system, that includes all blocks required for the data and file structure of a file system] of said file system." Id. (bracketing NetApp's proposed construction).

In support of its proposed construction, Sun points to NetApp's 7,174,352 patent ('352 Patent), which shares most of the inventors of the '292 patent and was filed roughly eight years after the '292 and '211 specifications were drafted. The '292 and '352 patents are continuations of patent application number 71, 643, which was abandoned. The '352 patent's specification concerns NetApp's WAFL file system which is disclosed in the '292 and '211 patents, and the '352 patent

incorporates the disclosures of the ’292 patent. NetApp acknowledged that the patents were “cousins.” The ’352 patent includes an express definition of consistent: “As used herein, the term ‘consistent,’ referring to a file system (or to storage blocks in a file system), means a set of storage blocks for that file system that includes all blocks required for the data and file structure of that file system.” ’352 Patent at 4:24-27 (Williamson Decl., Ex. D). “Thus, a consistent file system stands on its own and can be used to identify a state of the file system at some point in time that is both complete and self-consistent. Id. at 4:28-30. This definition is identical to Sun’s proposed construction, which now includes the “file” language. Sun’s definition, therefore, finds support in this related patent’s specification.

There are passages in the ’292 patent specifications that support NetApp’s construction and describe consistent states or consistency points that are rooted by a root inode. For example, the specification notes that “[c]hanges to the file system are tightly controlled to maintain the file system in a consistent state. The file system progresses from one self-consistent state to another self-consistent state. The set of self-consistent blocks on disk that is rooted by the root inode is referred to as a consistency point.” ’292 Patent at 4:9-13. Figure 16 and the accompanying description of a file system in a “consistent state” shows the consistent state of the system as being rooted by a root inode. Id. at 11:6-27 & Fig. 16. However, while consistency point and consistent state are closely related concepts, it is not entirely clear that the terms are identical. For example, “consistency point” is claimed in claim 4 of the ’292 patent, and “consistent state” appears in claim 8 of the ’292 patent. Sun notes that NetApp’s construction equating consistent states with consistency points cannot be correct, because the second consistent state is achieved while the file system is still at a first consistency point. However, as NetApp points out, the ’352 patent lends some support to NetAPp’s position, as it refers to a “state of the file system at some point in time that is both complete and self-consistent,” which appears to refer to both a consistent state and consistency point in a similar way. ’352 Patent at 4:24-37. Regardless of whether consistency state and consistency point are synonymous, however, NetApp’s proposed construction is too redundant of the surrounding claim language, as discussed above.

Finally, to the extent that NetApp’s construction incorporates its definitions of “file system”

United States District Court
For the Northern District of California

and "root inode" that the Court repeated above, this construction is similarly flawed. In sum, and for the reasons set forth above, the Court adopts Sun's construction and finds that the term means "a set of storage blocks for that file system that includes all blocks required for the data and file structure of the file system."

’715 Patent

The ’715 patent is directed to a technology for controlling data storage over arrays of disk drives. ’715 patent at 1:7-9. Specifically, the technology improves the read and write efficiency of a "Redundant Array of Inexpensive Disks" ("RAID array") in combination with a file system. RAID systems can provide greater storage capacity than a traditional single disk drive system because additional disk drives can be added to the array, and RAID systems have high data transfer rates. The techniques used in RAID systems generally allow data to be protected even when individual disks storing data experience failure, and when a single disk fails, it can be replaced without shutting down the entire system. The RAID array accomplishes this using a "stripe" technique. Data blocks are written across the array of disk drives in "stripes." The patent defines a "stripe" as including "one storage block on each disk drive in an array of drives in the system." Id. at 1:37-39. Essentially, data is written over an array of disks as a series of stripes, and at least one of these disks is used to store parity (also called redundancy data), and the others are used to store data. The parity information can be used to reconstruct a data file from a failed disk.

There is a conflict between optimizing the functionality of file systems and individual disks on the one hand, and optimizing the performance of a RAID on the other. File systems and disks are optimized by writing large amounts of related data contiguously on an individual disk. RAIDS, on the other hand, tend to be optimized by writing a full stripe all at once in order to minimize the amount of time spent reading data from disks in order to recalculate parity. However, when organized in this way, related data tends to be broken up into small pieces and performance suffers. The invention of the ‘715 patent is directed to optimizing both disk storage and parity recalculation. Specifically, the patent discloses a data structure, called an "association" for associating data blocks with storage blocks over the stripes. The multiple requests are ultimately written on multiple disks in the RAID in order to write whole stripes at a time.

14. "Associating the Data Blocks with One or More Storage Blocks Across the Plurality of Stripes as an Association"/"the Association to Associate the Data Blocks with One or More Storage Blocks Across the Plurality of Stripes"

NetApp's Proposed Construction	Sun's Proposed Construction
NetApp contends that neither phrase requires construction given their plain and ordinary meaning. In the event the Court construes the terms, NetApp proposes slightly different constructions for the phrases: "creating a data structure that relates data blocks to locations on more than one stripe," and "the data structure relating data blocks to locations on more than one stripe," respectively.	Sun contends both of these phrases are indefinite under 35 U.S.C. § 112 ¶ 2, and cannot be reasonably construed because they fail to particularly point out and distinctly claim the subject matter which the applicants regard as their invention. However, to the extent the Court construes the phrases, they mean "associating each data block with a respective one of the storage blocks across the plurality of stripes as an association."

To begin its analysis, the court first turns to the claims themselves. The first term appears in claims 21 and 52 of the '715 patent, and the second claim appears in claim 39 of that patent.

Claim 21 is for "a method for controlling storage of data, comprising:

receiving one or more write requests associated with data blocks; receiving topological information associated with storage blocks configured in a plurality of parallel stripes of a storage system;

associating the data blocks with one or more storage blocks across the plurality of stripes as an association;

and writing the data blocks, in response to the association, to the one or more storage devices in a single write request." '715 Patent at 21:40-52 (emphasis added).

Claim 39 is for a "storage system, comprising:

a file system, the file system to receive one of more write requests associated with data blocks;

a storage device manager, the storage device manager to generate topological information of storage blocks configured in a plurality of parallel stripes of one or more storage devices, and to send the topological information to the file system; and

an association generated in the file system, **the association to associate the data blocks with one or more storage blocks across the plurality of stripes** of the one or more storage devices, the association to be sent to the storage device manager, the storage device manager to

write the data blocks, in response to the association, to the one or more storage blocks as a single write request." '715 Patent at 22:25-40 (emphasis added).

Sun first argues that the claim is indefinite under 35 U.S.C. § 112 ¶ 2. A claim is indefinite under this paragraph, if "it would be apparent to one of skill in the art, based on the specification, that the invention set forth in the claim is not what the patentee regarded as his invention." Allen Eng'g Corp. v. Bartell Indus., Inc., 299 F.3d 1336, 1349 (Fed. Cir. 2002). A claim is also indefinite under that paragraph when it is "inherently inconsistent" or when its "legal scope is not clear enough that a person of ordinary skill in the art could determine whether a particular [product] infringes or not." Geneva Pharms., Inc. v. GlaxoSmithKline PLC, 349 F.3d 1373, 1384 (Fed. Cir. 2003).

Of course, claims are not indefinite merely because they present a difficult task of claim construction. Instead, "[i]f the meaning of the claim is discernible, even though the task may be formidable and the conclusion may be one over which reasonable persons will disagree," the claim is "sufficiently clear to avoid invalidity on indefiniteness grounds." Exxon Research & Eng'g Co. v. United States, 265 F.3d 1371, 1375 (Fed. Cir. 2001) (citations omitted). Proof of indefiniteness requires an exacting standard because claim construction often poses a difficult task over which "expert witnesses, trial courts, and even the judges of this court may disagree." Id. "Nevertheless, this standard is met where an accused infringer shows by clear and convincing evidence that a skilled artisan could not discern the boundaries of the claim based on the claim language, the specification, and the prosecution history, as well as her knowledge of the relevant art area." Halliburton Energy Servs. v. M-I LLC, 514 F.3d 1244, 1250 (Fed. Cir. 2008).

Sun argues that the phrase is indefinite because it is flatly inconsistent with and impossible under the teaching in the specification. The claim language requires "one or more storage blocks across a plurality of stripes," but under the invention, it is impossible for any "one" storage block to be "across a plurality of stripes." The specification provides that "a stripe includes one storage block on each disk drive in an array of disk drives." '715 Patent at 1:37-39. A storage block is a component of a single stripe, is located on a single disk, and does not traverse a plurality of stripes. Id. at 6:35-38 & Figs. 1A&B. Each square in Figure 1B is a separate storage block, and no one storage block is located in more than one stripe. See also id. at Fig. 8.

United States District Court
For the Northern District of California

NetApp concedes that one storage block cannot exist on multiple stripes at the same time. Yet NetApp's proposed construction recognizes that a single storage block only exists on one stripe, as NetApp expunges the infeasible "one or more" language from the claims so that "data blocks" are related to storage block locations on more than one stripe. The Court, however, cannot rewrite claims to preserve their validity. Allen Eng'g Corp., 299 F.3d at 1349 ("It is not our function to rewrite claims to preserve their validity."). NetApp argues that in the context of the specification, the claims mean only that the majority of the time, storage blocks will be written across more than one stripe. According to NetApp, the rare write request that is too small to occupy blocks on multiple stripes should be included in the claim. Including this single stripe scenario in the claim, however, contradicts the plain claim language, which indicates that one or more than one storage block will be across a *plurality* of stripes. While "[i]f one skilled in the art would understand the bounds of the claim when read in light of the specification, then the claim satisfies section 112 paragraph 2," Exxon Research & Eng'g Co. v. United States, 265 F.3d 1371, 1375 (Fed. Cir. 2001), the problem here is that NetApp's proposed construction both rewrites the plain language of the claims and contradicts the specification.

Nor is NetApp's extrinsic evidence particularly persuasive, even if it could trump the intrinsic evidence. NetApp's expert claims that "one of ordinary skill in the art would understand that the term 'one or more' covers a degenerate (abnormal) case only [in which the association is only a single storage block on a single stripe], because, in practical terms using more than one stripe, and therefore more than one storage block, is an important feature of the data structure . . ." Ganger Decl. ¶ 94. Thus, he concedes that using more than one stripe is an important feature. Dr. Ganger also argues that in the "degenerate case," even though the data block associated with a single storage block itself does not itself extend across a plurality of stripes, the storage blocks as a whole exist across a plurality of stripes, so one data block can be associated with one storage block across a plurality of stripes. But Dr. Ganger cites no intrinsic evidence in support of this convoluted and strained interpretation, which, as discussed above, contradicts the plain language of the claim. And, of course, Dr. Brandt disputes this conclusion. Hence, one of skill in the art would not know if a single storage block in one stripe would be within the claim scope or not. "That is the epitome of

indefiniteness." Geneva Pharms., Inc. v. GlaxoSmithKline PLC, 349 F.3d 1373, 1384 (Fed. Cir. 2003). Moreover, NetApp undermines its own argument by noting that the specification distinguishes prior art on the basis that the prior art does not generally carry out writes to multiple stripes. See, e.g., Ganger Decl. ¶ 95 ("Indeed, the specification frequently describes multi-stripe writes and distinguishes itself from the prior art on the basis of the prior art not performing writes to multiple stripes."); Ho Decl., Ex. L at 20 (Preliminary Feb. 22, 2005 Amendment) ("applicant goes one step further by mapping each data block of the single write request with a storage block across a plurality of stripes of the storage system before transmitting the buffered write request. . . ."). Consequently, it does not appear from the prosecution history that this "degenerate case," in which only one single storage block is associated with a single stripe, was contemplated by the invention.

In sum, because it is unclear (at best) to one of ordinary skill in the art as to whether the claim could encompass the so-called "degenerate case," and because NetApp's interpretation contradicts the plain meaning of the actual claim language, the claim is indefinite. Sun has shown by clear and convincing evidence that a skilled artisan could not discern the boundaries of the claim based on the claim language, the specification, the prosecution history, and knowledge of the relevant art area. Halliburton Energy Servs., 514 F.3d at 1250.

Because the claim is indefinite, the Court need not determine which of the parties' proposed constructions is proper. Sun's construction would have required a one-to-one correspondence between each data block and an associated storage block, and NetApp's would not. As this issue relates to the indefiniteness issue discussed above, however, the Court notes that it agrees that the intrinsic evidence requires a one-to-one correspondence. The abstract clearly states that "[e]ach data block of the data to be written is associated with a respective one of the storage blocks. . ." The specification states that the block layout information "associates each data block of the buffered write requests 41 with a storage block in a group of storage blocks." '715 Patent at 9:16-19; see also id. at 13:37-39 (referring to one-to-one association with data blocks); 13:20-22 ("each data block is stored at its associated storage block in the group"); 13:2-5 ("unallocated storage blocks are selected in the array 20A for storage of a corresponding number of data blocks associated with buffered write requests."). The specification, therefore, indicates that each data block is associated with a

respective one of the storage blocks. NetApp does not contradict this evidence, nor does it point to any examples in the specification describing something other than a one-to-one correspondence. Therefore, the specification strongly supports Sun's construction.

The prosecution history supports this requirement as well, as the patent applicant distinguished prior art on the basis that this invention "associat[ed] each data block with a respective one of the storage blocks, for transmitting the association to a storage device manager for processing of the single write transaction." Williamson Decl., Ex. F at 17; see also id. at 17-18 ("Applicant goes one step further by associating each data block of the single write request with a storage block of the storage system."); Ex. H at 20 (distinguishing prior art after subsequent rejection, noting that applicant claimed novel "mapping each data block with a respective one of the storage blocks across a plurality of stripes"); Ex. I at 15 ("associating each data block with a respective one of the storage blocks"). In allowing the claims, the examiner noted that the prior art did not teach buffering write requests, "associating each data blocks to be stored with a respective one of the storage blocks across a plurality of stripes for a single write operation." Williamson Decl., Ex. J at 2.[11]

However, as discussed above, the phrase "associating the data blocks with one or more storage blocks" in the claim language indicates that there is not a one-to-one correspondence, as the claim language requires multiple data blocks, but allows only one storage block in certain situations. Because this claim language cannot be reconciled with the specification, which clearly requires a one-to-one correspondence, this finding that there is a one-to-one correspondence in this claim term further supports the finding that the claim is indefinite. In sum, and for the reasons discussed above, the Court finds that the claim is indefinite under 35 U.S.C. § 112 ¶ 2.

III. CONCLUSION

In accordance with the foregoing, and for the reasons discussed above, the court construes the parties' disputed terms as follows:

---

[11] Because the claim is indefinite, the Court need not resolve the parties' other disputes regarding construction of the term. However, the Court briefly notes that NetApp does not provide any support for its use of the term "locations" in lieu of "storage blocks," and replacing this claim language with broader language seems improper. NetApp also argues that its construction makes clear that the data structure called an "association" associates data blocks with storage blocks, while Sun's construction does not. At the hearing, however, Sun conceded that should the term be construed, it did not object to the definition "data structure" for the term "association."

1. "Domain name" means "a unique name that has a unique numerical IP address associated with it."

2. "Server identification data" means "information that uniquely identifies one server from other servers, that is more human user-friendly than the server's IP address, domain name, URL, or hypertext link."

3. "In response to writing a data record to said one redundancy group"/"Responsive to writing a data record to one of said redundancy groups" means "after and in reaction to the writing of a data record to a single redundancy group."

4. "In response to the receipt of a stream of data records from said data processor"/ "Responsive to the receipt of a stream of the data records from said data processor" means "after and in reaction to the receipt of data records from a processor."

5. "Completing [a] Write Operation Within [a] Local Processing Node" / "Completing [a] Write Operation With Respect to [a] Processor" is not yet construed. The parties shall submit an agreed upon construction no later than **September 19, 2008**, as discussed above**.**

6. "Portion [of a] Communication" means "part of the data stream, where the part includes elements of the data stream."

7. "Element [of a] Communication" means "a constituent part of a portion."

8. "Non-volatile storage means" is a means-plus-function limitation governed by 35 U.S.C. § 112, ¶ 6, wherein the function is storing blocks of data of a file system so that the data is not lost in the absence of power, and the structure is a disk or disk arrays that are suitable for storing 4 KB blocks.

9. "Meta-data for successive states of said file system" tentatively means "information describing a copy of the structure of the active file system (such as a copy of a root inode) at a series of successive points in time." The parties are to meet and confer about the propriety of this proposed construction and may submit a different agreed upon construction that comports with the guidance in this order no later than **September 19, 2008.**

10. "File system information structure" means "data structure that contains the root inode of a file system in a fixed location on disk."

11. "On-disk root inode" means "the index node data structure stored in a fixed location on disk that roots a set of self-consistent blocks on the storage system that comprise the file system," and "incore root inode" means "the copy of the on-disk root inode that is contained in memory; this copy need not be in a fixed location in memory."

12. "Pointing directly and indirectly to buffers in said memory and a second set of blocks on said storage system" means "pointing directly and indirectly to buffers in said memory and pointing directly and indirectly to a second set of blocks on said storage system."

13. "Consistent state"/"State of a file system" means "a set of storage blocks for that file system that includes all blocks required for the data and file structure of the file system."

14. "Associating the Data Blocks with One or More Storage Blocks Across the Plurality of Stripes as an Association"/"the Association to Associate the Data Blocks with One or More Storage Blocks Across the Plurality of Stripes" are indefinite phrases under 35 U.S.C. § 112 ¶ 2

All of these terms shall be so construed where they appear in the claim preambles, as well as in the body of the claims.

IT IS FURTHER ORDERED that a case management conference is set for **October 3, 2008** at 3:00 p.m. to discuss setting further dates in the case. The parties shall file a joint case management statement no later than September 26, 2008. Proposals for scheduling summary judgment motions should take into account the timing of the claim construction in related case number 08-1641, and whether there should be a limit on the number of such motions and whether such motions should be staggered and prioritized.

IT IS SO ORDERED.

Dated: September 10, 2008

Elizabeth D. Laporte

ELIZABETH D. LAPORTE
United States Magistrate Judge